Applied Soft Computing Journal 180 (2025) 113421

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Federated semi-supervised learning with contrastive representations against noisy labels

Wenjie Mao [a], Bin Yu [a], Yihan Lv [a], Yu Xie [b], Chen Zhang [a,*]

[a] School of Computer Science and Technology, Xidian University, Xi'an 710071, China
[b] School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China

ARTICLE INFO

Keywords:
Federated semi-supervised learning
Contrastive learning
Label noise
Image classification

ABSTRACT

Federated semi-supervised learning presents a pragmatic scenario wherein a centralized model is trained utilizing a server with access to labeled data, while participating clients lack any labeled data. In this context, the inaccuracy of real-world labels on the server available for training poses a huge challenge to the federated semi-supervised learning. These inaccuracies can have a detrimental impact on the overall performance of the system and impose limitations on its use. In this paper, we propose a novel Federated Semi-supervised learning framework with Contrastive Representations, called FedCR, with the aim of addressing the aforementioned ubiquitous problems in the field of image classification tasks. Firstly, our approach employs contrastive representation learning to build memory representations of images, which can learn an image's general features from an augmented view without relying on negative pairs and prevent the model from memorizing noise. Then we take a cautious approach during model updates to prevent any potential leakage to ensure the privacy and security of the clients' information. Additionally, for the sake of improving robustness of the model, a contrastive regularization function is applied to preserve information connected to true labels while filtering out information associated with wrong labels. Furthermore, we mitigate the negative impact of mislabeled data during supervised learning by utilizing an improved cross-entropy loss function. Extensive experiments on prevalent datasets for image classification tasks show that the proposed method surpasses previously established state-of-the-art federated semi-supervised learning algorithms and efficiently alleviates the issue of model over-fitting to erroneous labels, especially when label noise is present.

1. Introduction

Federated learning [1,2] has emerged as a prominent research domain, offering the unique capability of training machine learning models across decentralized datasets while simultaneously preserving data privacy. While various federated learning methods have been designed and successfully applied in different domains such as images analysis [3], objection detection [4,5], and Internet of Things (IOT) applications [6], most existing approaches critically operate under the unrealistic assumption of considering only the supervised learning setting, where local private data is fully and correctly labeled. Although this assumption simplifies the problem formulation, it remains largely inapplicable in numerous real-world scenarios where data labeling requires specialized expertise and substantial user motivation.

To address this limitation, federated semi-supervised learning (FedSSL) has garnered increasing attention from the academic and research communities in recent years. In this novel setup, clients predominantly possess unlabeled data, while the central server maintains a small subset of labeled data, known as labels-at-server scenario (see Fig. 1). Extensive researches have increasingly explored methodologies to synergize federated learning principles with traditional semi-supervised techniques, with the objective of effectively leveraging unlabeled data for enhancing the advancement of the global model. For example, Jeong et al. [7] focused on studying the inter-client consistency loss through the enforcement of identical class labels on both the augmented and original instances by way of consistency regularization. Besides, they utilized a unique pseudo-labeling methodology which refers to as the agreement-based pseudo label. The study in [8] involves the consistent regularization loss, for which group normalization is used in lieu of batch normalization to decrease gradient diversity. Additionally, a grouping-based model averaging technique supplants the federated averaging (FedAvg) [9,10] algorithm. Wei et al. [11] took into account the fairness of the federated semi-supervised learning scenario by globally balancing the number of unlabeled samples. Their

[*] Corresponding author.
E-mail address: zhangc@xidian.edu.cn (C. Zhang).

https://doi.org/10.1016/j.asoc.2025.113421
Received 27 March 2024; Received in revised form 3 April 2025; Accepted 1 June 2025
Available online 14 June 2025

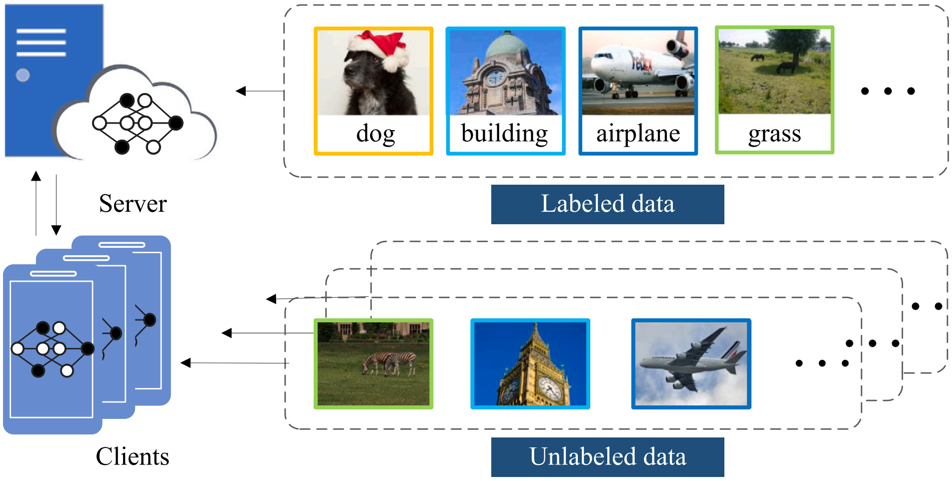


Fig. 1. Illustration of the labels-at-server scenario in federated semi-supervised learning setting. Labeled and unlabeled data are distributed on the server and clients respectively. Furthermore, the data across different clients may be non-IID.

method can dynamically adjust the data threshold to generate better pseudo labels.

Despite these advancements, two practical issues have failed to be addressed in current FedSSL methods. First, the performance of semi-supervised learning remains constrained by fundamental architectural limitations. The performance of performing semi-supervised learning in a federated learning environment is limited. In conventional semi-supervised learning scenarios, a single model simultaneously learns from both labeled and unlabeled data, with unlabeled data providing valuable distribution insights and the labeled data acting as a constraint to prevent error propagation. In other words, effectively constraining the model's updates through labeled data to prevent it from memorizing errors that may have originated from unlabeled data. However, in federated semi-supervised learning settings, labeled and unlabeled data are usually distributed in disjoint locations. As a result, the labeled data cannot promptly provide correct constraints for the global model. Moreover, the incorporation of unlabeled data during the training process not only fails to impart supplementary distributional information to the model, but also results in the accumulation and error propagation from the unsupervised learning process. As a consequence, the overall performance of the global model is compromised. Another pivotal issue is that in real-world scenarios, data labels are inherently imprecise. Acquiring accurately labeled datasets in federated semi-supervised contexts is a formidable, labor-intensive and costly endeavor [12]. To minimize the labor and cost associated with labeling, a viable direction is to rely on manually annotated datasets obtained from crowd sourcing platforms like Amazon Mechanical Turk and CrowdFlower. Consequently, errors are bound to occur during the labeling process. Once the such noisy labels are incorporated into model training, cross-entropy loss mechanisms can inadvertently memorize these inaccuracies, precipitating substantial performance degradation [13,14]. Therefore, designing a noise-robust approach is essential to mitigate the adverse effects of erroneous labels in FedSSL.

In this work, we present a new simple and efficient federated semi-supervised learning framework FedCR, which is designed to extract robust and critical knowledge from noisy labels. To accomplish this objective, we initially employ a novel siamese-like network module for instance-based contrastive representation learning, which is effective in preventing deep networks from over-fitting noisy labels. This module consists of an online network updated by global parameters and a target network updated locally via exponential moving average (EMA). To ensure client data privacy, only the online network's parameters are exchanged, while the target networks are trained locally based on the server's online model to accommodate the federated learning environment. This update strategy also mitigates the challenge of non-independently identically distributed (non-IID) client data. As proper labeling information is crucial in correcting positive comparisons made by similar instances, FedCR utilizes the gradient variation difference between the correct and wrong comparisons to identify the correct labeling information from noisy labels. This mechanism effectively minimizes the negative impact of noisy labels. Moreover, the cross-entropy loss during the supervised learning process may lead to the model exhibiting over fitting with error labels, and it tends to memorize samples with incorrect labels. In an attempt to counter this negative impact, we suggest the integration of the mean absolute error with the cross-entropy loss function during supervised learning. This new loss promotes rapid convergence while bolstering the robustness against noise, ensuring better performance under noisy label conditions. Fig. 2 illustrates a toy example of our method for dealing with noisy labels. Extensive experiments on several natural image classification datasets demonstrate the effectiveness of our methodology. Overall, our main contributions are summarized as follows:

- We propose a novel and robust federated semi-supervised learning framework incorporating instance-based contrastive representation learning with a Siamese network and a privacy-preserving update strategy, engineered to mitigate the adverse effects of noisy labels.
- We design a hybrid loss function integrating a new contrastive regularization and an improved cross-entropy loss to optimize supervised and unsupervised learning, promoting rapid convergence and enhancing overall performance in non-IID environments.
- Comprehensive experiments on benchmark datasets demonstrate that FedCR significantly outperforms the other state-of-the-art federated semi-supervised learning methods, achieving superior performance under varying levels of label noise conditions.

The rest of this paper is organized as follows. We sketch the related works in Section 2. Section 3 formally defines the problem and details the proposed framework. The extensive experimental results are presented and analyzed in Section 4. Finally, in Section 5, we conclude the work and outline future research directions.

2. Related works

2.1. Federated learning with non-IID

Federated learning is a decentralized machine learning paradigm that facilitates the collaborative training of a global model across

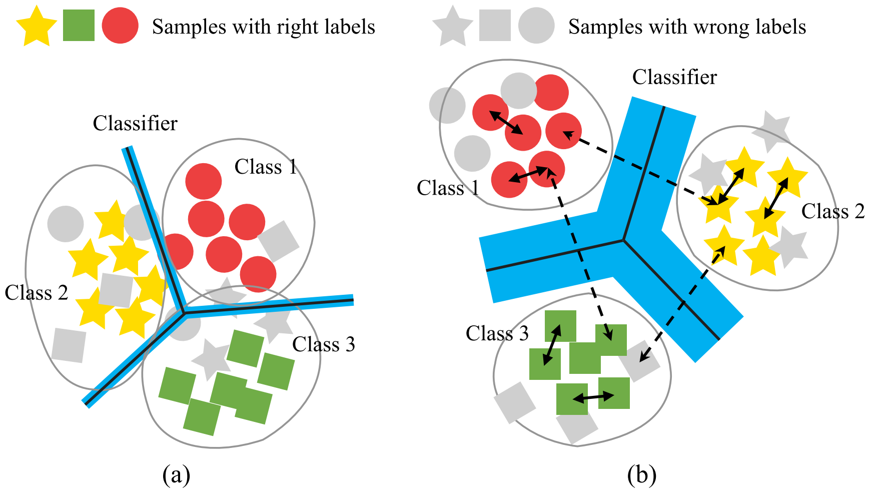


Fig. 2. An illustration of classification with noisy data by traditional methods and the proposed method. The black straight line is the best classifier learned after model training. (a) Neither processing noisy labels nor applying contrast loss constraints. (b) Our method. The goal is to distinguish different categories better. Samples of the same category are grouped into one cluster, and clusters are isolated from each other.

multiple clients without sharing the local data. A critical challenge in this setting is statistical heterogeneity, commonly referred to as non-IID data, which significantly impacts the learning performance of parametric models and often leads to global model divergence [15]. Numerous studies have sought to address this challenge, particularly in supervised federated learning scenarios [1,16–19]. One of the foundational approaches, FedAvg [1], demonstrates its capacity to mitigate the effects of non-IID data. Building on this, subsequent research can be broadly categorized into three strategies: (1) Data-based approaches: These include data-sharing mechanisms [20–22] and various data augmentation techniques [23,24] to reduce statistical discrepancies across clients. (2) Algorithm-based approaches: Methods such as personalized federated learning [25–27] aim to adapt global models to local tasks, thereby improving performance on heterogeneous data. (3) System-based approaches: Many studies explore client clustering to create multi-center frameworks, grouping clients into clusters to address non-IID distributions [28,29]. While these strategies have proven effective in supervised federated learning, they face significant limitations in federated semi-supervised learning settings. The lack of labeled data on clients and the scarcity of labeled samples globally hinder the application of these methods, necessitating new approaches tailored to semi-supervised environments.

2.2. Federated semi-supervised learning

Federated semi-supervised learning (FedSSL) extends federated learning to scenarios where labeled data is meagre, either on clients or the server. Three main settings have been extensively studied: (1) Labels-at-Client [7,8,30,31]: where local clients possess partially labeled data; (2) Labels-at-Partial-Clients [32–39]: where few clients are fully labeled, and the rest hold only unlabeled data, and (3) Labels-at-Server [7,8,40–45]: where fully labeled data resides on the central server, while local clients contain only unlabeled data. This setting, the most challenging, is the focus of our work due to its broader applicability.

Recent studies have proposed various strategies to address these scenarios: FedConsist [32] and FedIRM [33] utilize pseudo-labeling to generate virtual labels but struggle to generalize under non-IID data distributions CBAFed [35] addresses class imbalance and catastrophic forgetting through class-balanced adaptive pseudo-labeling. RSCFed [34] proposes consensus via random client sub-sampling, though it relies on standard regularization, making it susceptible to non-IID effects. Qiu et al. [39] developed a federated pseudo-labeling strategy using embedded knowledge from labeled clients. FedAnchor [44] introduces a label contrastive loss based on cosine similarity to mitigate confirmation bias and overfitting. FedTriNet [46] employs three networks and a dynamic quality control mechanism to produce high-quality pseudo labels for unlabeled data. $(FL)^2$ [45] combines adaptive threshold pseudo-labeling, consistency regularization, and state-aware aggregation to reduce confirmation bias. Semi-HFL [43] is a heterogeneous FL framework based on semi-supervised learning, used to address the challenges of resource heterogeneity and unlabeled data. SemiFL [42] utilizes alternate training that combines FL with semi-supervised learning by fine-tuning the global model with labeled data and generating pseudo labels using the global model. Despite these efforts, pseudo-labeling approaches often degrade model performance due to noisy labels. Unlike these methods, our approach avoids pseudo-labeling entirely, instead leveraging contrastive learning to amplify gradient updates for correct pairs, enabling robust processing of unlabeled noisy data.

Additionally, several contemporary studies explore complementary aspects of FedSSL. For instance, FedMatch [7] is widely recognized as a benchmark method in this regard, as it improves upon naive combinations of FL and semi-supervised learning to enforce inter-client predictions consistency. SSFL [8] identifies gradient dissimilarities across clients and proposes mitigation strategies. Besides, Wei et al. [11] examine fairness issues, alleviating accuracy imbalances caused by non-IID data. DCCFSSL [38] presents a dual class-aware contrastive module to improve global model performance when only partial clients have labeled data. ProtoFSSL [31] leverages prototypical networks but relies heavily on labels and pseudo labels, neglecting noise in labeled data. While these approaches mark significant progress, their reliance on direct combinations of federated and semi-supervised learning often limits their performance and resilience to label noise. Furthermore, the limited availability of labeled data, often accompanied by inaccuracies, hinders their effectiveness. To address these challenges, our focus is on the third category, aiming to comprehensively exploit the abundance of unlabeled data alongside noisy labeled data to enhance federated learning. We introduce a noise-robust contrastive module, advancing the state-of-the-art in FSSL by improving performance under realistic conditions with noisy and scarce labels.

2.3. Contrastive learning

Contrastive representation learning has emerged as a highly effective paradigm in unsupervised learning, achieving notable success in natural language processing and visual representation learning [47–49]. Unlike supervised methods, contrastive representation learning operates by comparing input samples to learn meaningful representations. Several classical approaches have demonstrated its efficacy: SimCLR [50] uses data augmentation to generate positive pairs and treats all other samples in a batch as negative examples, leveraging negative sampling to enhance representation learning. Moco V2 [47] introduces a momentum encoder and a dynamic queue to manage negative samples, enabling selection from the entire training set. Positive-pair-only methods, such as BYOL [51] and SimSiam [52], avoid using negative examples entirely, relying on self-supervised learning techniques to achieve competitive results. While these methods have made significant progress, they only consider centralized settings in unsupervised scenarios and are not directly applicable to federated settings.

Recent efforts have adapted contrastive learning for federated learning (FL) to address the lack of large labeled datasets on clients [53]. For instance: FedU [54] develops an communication-efficient mechanism by aggregating online encoders. SSFL [55] extends unsupervised training to personalized FL frameworks L-DAWA [56] introduces layer-wise divergence aware weight aggregation to mitigate client bias. Fedutn [57] employs aggregated online networks for updating target networks in a self-supervised framework. FLPD [22] uses prototype similarity distillation based on publicly labeled datasets, but the reliance on consistently available public data limits its practicality. FedU2 [58] reduces representation collapse entanglement in federated unsupervised learning. FedMKD [59] proposes a multi-teacher knowledge distillation framework to learn global class representations from heterogeneous clients. These methods have focused on enhancing unsupervised representation learning performance in federated settings. However, they overlook scenarios where the server possesses labeled data—a critical consideration in our research. In this work, we propose a novel contrastive module specifically tailored for FedSSL with noisy labels. Our approach extends traditional contrastive paradigms by addressing label noise while effectively leveraging the labeled data available at the server. This robust design not only improves representation learning in federated semi-supervised settings but also remains adaptable to other contrastive learning frameworks, showcasing its versatility and potential for broader applications.

2.4. Learning from noisy label

Acquiring large-scale labeled training data often relies on web services to reduce costs associated with manual labeling. However, this approach frequently introduces label noise. Even experienced domain experts may face challenges in accurately labeling complex data, which can be further exacerbated by adversarial manipulations such as label-flipping attacks [60]. Studies indicate that deep neural networks tend to memorize noisy labels, leading to overfitting and diminished generalization performance [13].

To address the issue of noisy labels, a key research direction has been the development of noise-robust loss functions [61]. (1) Cross-entropy-based methods: many approaches adapt cross-entropy loss to prioritize learning from clean data while mitigating the impact of noisy data. For instance: Zhang et al. [62] integrated the mean absolute error loss and the cross-entropy loss. Similarly, the symmetric cross-entropy loss [63] combines a noise-robust reverse cross-entropy loss with standard cross-entropy loss. (2) Negative learning techniques, such as those in [64,65], offer general strategies that can complement any classification loss function. (3) Contrastive regularization methods: these approaches focus on learning self-supervised representations without labels, effectively mitigating noisy label impact [66,67].

Other works target specific mechanisms to separate clean and noisy samples. For example, CNLL [68] employs semi-supervised continuous noisy label learning, relying on a sample separation mechanism. However, this method struggles in federated learning (FL) scenarios due to significant client-level data heterogeneity, which complicates accurate sample separation. In the FL context, noisy labels pose unique challenges due to decentralized and non-IID data distributions: SsCL [69] combines contrastive loss in self-supervised learning with cross-entropy loss in semi-supervised learning, but its efficacy diminishes under FL settings with limited labeled data and pervasive noise. Huang et al. [70] enhanced representation and label correction by supervised contrastive learning, yet similar limitations emerge in FL environments. In addition to the research on solving class-conditional noise in FL [71–74], FedBeat [75] uses Bayesian ensemble-assisted transfer matrix estimation to address instance-dependent noise problem. Despite these advances, existing methods often struggle with federated semi-supervised settings, where labeled data is both scarce and noisy. To overcome these limitations, we propose a novel contrastive regularization function with an enhanced cross-entropy loss function, aiming to improve model robustness and overall performance in decentralized settings with noisy labels, addressing key challenges in FedSSL.

3. Methodology

This section first presents the problem statement, then provides an overview of our framework (see Fig. 3). Finally, we describe its core components in detail in the remaining subsections.

3.1. Problem statement

Following the common federated learning setting, our method considers the classification problem with a single server and total N clients. This article mainly focuses on the labels-at-server scenario, in which the data obtained at clients is fully-unlabeled.

Let $\mathcal{X}$ be the sample space and $\mathcal{Y} = \{1, \dots, C\}$ be the label space. Assume there is a dataset $\mathcal{D}_s$ consisting of s labeled training data in server. In an ideal scenario, $\mathcal{D}_s = \left\{(x_i, y_i)\right\}_{i=1}^{s}$, where each $(x_i, y_i) \in (\mathcal{X} \times \mathcal{Y})$ and y_i is the ground truth. To take the effect of noise into account, we denote $\mathcal{D}_s$ with label noise by $\mathcal{D}_s = \left\{(x_i, y_i')\right\}_{i=1}^{s}$, where y' are the noisy labels with respect to each sample because the noise is defined to be class-dependent in general, so that:

$$p(y_i' = l | y_i = j, x_i) = p(y_i' = l | y_i = j) = r_{lj}, \tag{1}$$

r is the noise rate, which is uniform with noise. In the case where l and j are members of the identical class, $r_{lj} = 1 - r$; when l and j are affiliated with distinct classes, $r_{lj} = \frac{r}{c-1}$. For each client $k \in \{1, \dots, N\}$, we denote its unlabeled dataset $\mathcal{D}_k = \left\{(x_i)\right\}_{i=1}^{n_k}$. The number of labeled data in server is considered to be much smaller than the total number of unlabeled data on the clients, i.e., $|\mathcal{D}_s| \ll \sum_{k=1}^{N} |\mathcal{D}_k|$. Note that the unlabeled data distributions at different clients/users may not follow the IID assumption. In each round of communication, K clients are randomly selected ($K \ll N$). Our goal is to derive a good global model by utilizing both labeled and unlabeled data in a decentralized setting to perform the classification task.

3.2. Overview

FedCR employs an iterative approach where there is an exchange of local–global models between the server and clients. In contrast to the conventional supervised federated learning pipeline, our training commences with an emphasis on server-side operations. This is due to the fact that our model can only acquire classification knowledge related to labels exclusively through server-side supervised learning, thus offering significant benefits for data clustering. We divide the global network into two sub modules, namely classification module and contrastive representation module. The former branch mainly conducts supervised learning, while the latter performs unsupervised contrastive learning. Our code is available at: code.

The proposed FedCR framework is illustrated in Fig. 3. The model training process comprises several communication rounds and each round consists of several training epochs on both the server and clients. In the initial round, we pre-train a global model $\{\theta, \xi, \sigma\}$ on the server

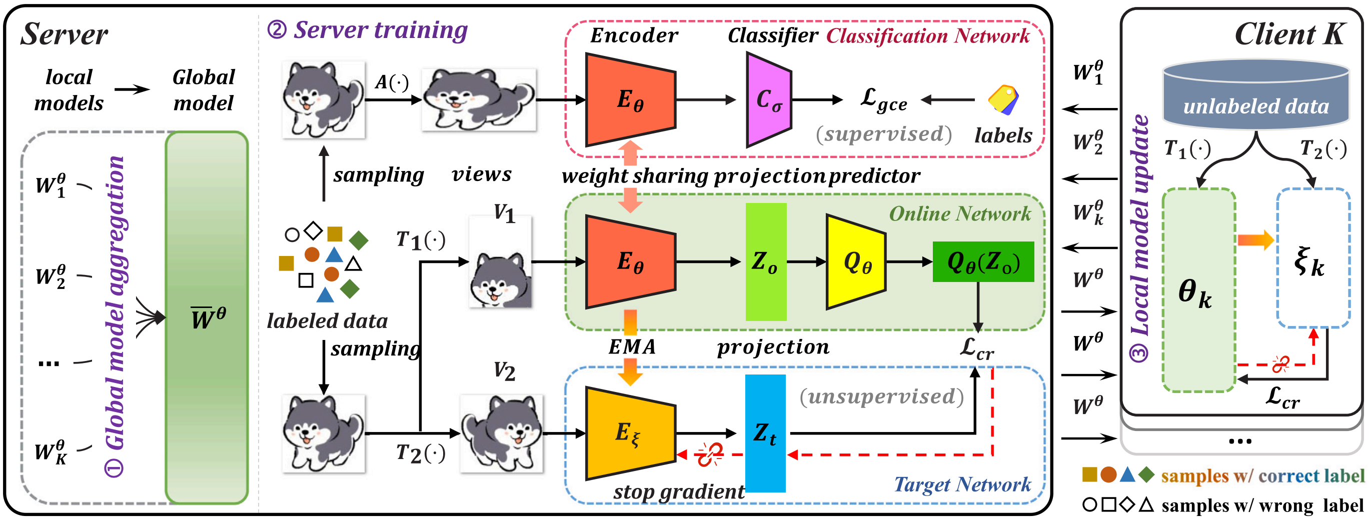


Fig. 3. An overview of the proposed FedCR. On the server-side, the global model is optimized by the generalized cross-entropy loss L_{gce} and contrastive regularization function L_{cr} utilizing inaccuracy labeled data. And the parameters of online network (θ) are further distributed to each client for local update of the next round. Each client k updates the contrastive representation network with its unlabeled data and uploads θ_k to the server. The server aggregates the parameters of the contrastive learning network on all selected clients and then updates the global model.

side utilizing available noisy labels. Specifically, this pretraining involves the training of sub-networks within the contrastive representation module, which comprises an online network, characterized by its weights denoted as θ, and a target network, which employs a different set of weights ξ. Furthermore, the classification module σ is also subject to this pretraining process. In the subsequent rounds of federated semi-supervised learning, we perform the following steps. (1) Broadcasting: for the rth round, the server broadcasts the global online network θ to random K clients. (2) Local model update: the client performs unsupervised learning to train the local contrastive representation module and uploads the online network to the server. (3) Global model aggregation: the server aggregates clients' online networks for refresh the global online network. (4) Server training: the server updates global model $\{\theta, \xi, \sigma\}$. These steps are repeated for R_g rounds.

3.3. Contrastive representation module

The effectiveness of contrastive representation learning has been demonstrated through its extensive application in various tasks [76,77]. The core objective of contrastive representation learning is to reduce the distance between representations of different augmented views of the positive samples. By learning to bring representations of positive samples closer together, the model can better capture and understand the underlying structures and patterns within the data. Moreover, self-supervised contrastive representation learning has been both empirically and theoretically confirmed as a method that effectively prevents models from memorizing incorrect labels [78].

In our approach, we utilize a curated contrastive representation module to leverage both noisy labeled data and unlabeled data for training on the server side and client side. This approach enables the model to identify common features across similar instances and differentiate dissimilar instances, thereby enhancing robustness to noisy labels and improving the classification performance of supervised learning. The proposed contrastive representation module comprises an online network θ, which includes an encoder E_θ and a predictor Q_θ, as well as a target network ξ, which consists of an encoder E_ξ. Note that each encoder contains a backbone and a projector, which are represented as B_θ and G_θ for the online network and B_ξ and G_ξ for the target network. See Fig. 4 for the internal structure of the encoder. In the training process, the online network shares the same encoder weight parameters with the classification network.

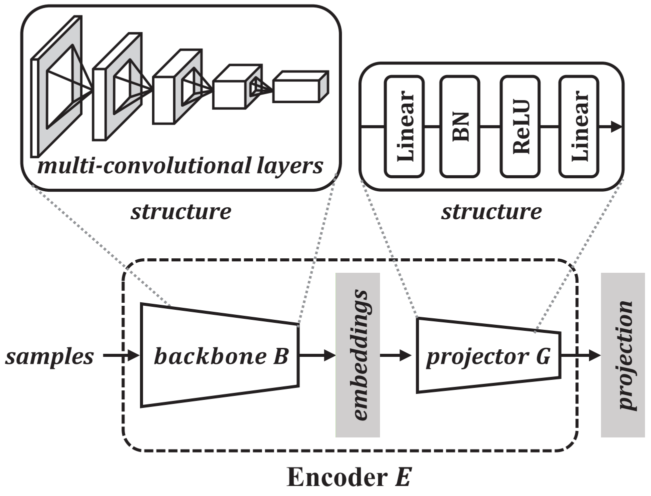


Fig. 4. The detailed flowchart from the input to the output of the encoder module. The encoder includes a backbone network that outputs low-dimensional embedding and a projector that outputs high-dimensional projection, both of which serve as feature extraction networks. It is worth mentioning that the projector is a multi-layer perceptron with the same structure as the predictor Q_θ.

Given an input image x from a batch of dataset, two transformations T_1, T_2 are randomly sampled from a transformation set $\mathcal{T}$ (including several available modes, such as cropping, flipping, color jitter, and random gray-scale) to generate two distinct views $V_1 = T_1(x)$, $V_2 = T_2(x)$. In order to learn useful representations, for both the online and target networks, we use backbones B_θ and B_ξ to get the embedding, and then connect them to projectors G_θ and G_ξ to map inputs to the representation space, which output projections Z_o and Z_t, respectively (See Fig. 3). The difference is that the online network is finally connected with a predictor Q_θ to obtain the prediction $Q_\theta(Z_o)$. This asymmetry between the online network and the target network can prevent model from collapsing and enhance the efficacy of representation learning performance. In addition, during the training phase, we set stop-gradient after the last module of the target network (i.e., the projector G_ξ), to block the back propagation of the gradient, which helps stabilize the training process by avoiding the issue of gradients vanishing or trivial constant solutions, thereby ensuring that the gradients remain within a manageable range, facilitating more stable and efficient model training. Specifically, it prevents gradients from flowing through the target network back into the online network, thereby avoiding gradient loops that could destabilize training or lead to representation collapse, where the model outputs trivial solutions like constant vectors. By halting gradient propagation through the target network, stop-gradient ensures that only the online network is directly optimized, while the target network serves as a stable teacher providing meaningful supervisory signals. This separation of optimization paths enhances training stability, prevents interference between the networks, and favors the learning of diverse and meaningful representations without relying on negative samples. Thus, stop-gradient is essential for maintaining the effectiveness and robustness of self-supervised learning methods. Therefore, the parameters of the target network can be updated by a momentum mechanism instead of using back-propagation. Specifically, the weights of the target network ξ can be calculated by the exponential moving average (EMA) of the online parameters θ. Formally, given a target decay rate $\tau \in [0, 1]$, it updates as follows:

$$W^{\xi} \leftarrow \tau \cdot W^{\xi} + (1 - \tau) \cdot W^{\theta}, \tag{2}$$

where τ is used to balance the influence of current parameters and historical values. Through EMA, the value of historical records is comprehensively considered to make the parameter update of the target network more stable and prevent data jitter.

In order to improve the convergence of the model, the contrastive representation learning module usually adopts the mean square error (MSE) function. However, the sensitivity of the MSE loss to outliers may compromise the predictive accuracy of other non-outlying data points. Additionally, if the data on the server contains noise, the MSE loss can lead to a decrease in the distinguishability of representations from different classes, thereby resulting in a potential negative impact on the global model's performance. To address this issue, we first consider the following loss for all images pairs $(x_j,\ x_k)$ belong to the same class:

$$\begin{aligned}\mathcal{L}_c &= -\sum_{(x_j,x_k)\in\mathcal{X}} \ell(x_j, x_k), \\ &= -\sum_{(x_j,x_k)\in\mathcal{X}} [(\langle \tilde{q}_j, \tilde{z}_k\rangle + \langle \tilde{q}_k, \tilde{z}_j\rangle) \times \mathbb{1}\{y_j = y_k\}],\end{aligned} \tag{3}$$

where $\tilde{q}$ and $\tilde{z}$ are normalized unit vectors, $\tilde{q} = Q(Z_o)/\|Q(Z_o)\|_2$, $\tilde{z} = Z_t/\|Z_t\|_2$. Q is the predictor within the online network, $\langle\cdot,\cdot\rangle$ denotes cosine similarity. $\mathbb{1}\{\cdot\}$ indicates the indicator function. From Eq. (3), it can be observed that the representations of x_j and x_k are brought closer when $y_j = y_k$. However, if there is noise in the labels, the result obtained by the indicator (i.e., $\mathbb{1}\left\{y_j = y_k\right\}$) becomes unreliable when minimizing $\mathcal{L}_c$. Since deep models initially fit samples with clean labels, and the probability outputs are greater than those of samples with incorrect labels [79]. Given the contrastive threshold η, one simple solution is to select a more reliable criterion $\mathbb{1}\left\{p_j^{\top} p_k \geq \eta\right\}$ to replace $\mathbb{1}\left\{y_j = y_k\right\}$, where p is the probabilistic output produced by linear classifier C_σ on the representation of images x, i.e., $p = softmax(C_\sigma(G_\sigma(B_\sigma(x))))$. However, this change only aids in the early stages of contrastive representation learning [66]. One possible reason is that in the early stages, the prediction q of two samples from the same class are more similar than those of two samples from different classes. After this, examples with noisy labels lead the learning process because the gradient magnitude from the wrong contrast pairs exceeds those from the correct pairs. In other words, the model gradually tends to adapt to the mislabeled data, leading to a decline in model performance.

To acquire more reliable representations, a regularization technique is applied to impose constraints on the model. When images x_j and x_k belong to the same class, it means that their semantic information is consistent, and the rich visual features extracted from them are specific to that particular class. On the contrary, if two samples belong to different categories, their representations reflect significant differences in semantic information. Therefore, we introduce a new contrastive regularization function that can be used to constrain image features, resulting in the effective minimization of the distance between images from the same class:

$$\begin{aligned}\mathcal{L}_{cr} = \sum_{(x_j,x_k)\in\mathcal{X}} l(x_j, x_k) = \sum_{(x_j,x_k)\in\mathcal{X}} \left[\log\left(1 - \langle \tilde{q}_j, \tilde{z}_k\rangle\right) + \log\left(1 - \langle \tilde{q}_k, \tilde{z}_j\rangle\right)\right] \\ \times \mathbb{1}\left\{p_j^{\top} p_k \geq \eta\right\}.\end{aligned} \tag{4}$$

Here, we perform a gradient analysis on sample-level constraint function $\ell'(x_j, x_k)$, which can be formulated as follows:

$$l(x_j, x_k) = \log(1 - \frac{Q(Z_o)^j \cdot Z_t^k}{\|Q(Z_o)^j\|_2 \cdot \left\|Z_t^k\right\|_2}) + \log(1 - \frac{Q(Z_o)^k \cdot Z_t^j}{\|Q(Z_o)^k\|_2 \cdot \left\|Z_t^j\right\|_2}) \tag{5}$$

Due to the action of the indicator function, we only need to consider the case where $p_j^{\top} p_k \geq \eta$ is true, otherwise the function value is 0. For simplicity, we denote $q_j = Q(Z_o)^j$ and assume the predictor Q is an identity mapping layer (i.e., $q_j = z_j$), we have:

$$\begin{aligned}l(x_j, x_k) &= \log(1 - \frac{q_j \cdot Z_t^k}{\left\|q_j\right\|_2 \cdot \left\|Z_t^k\right\|_2}) + \log(1 - \frac{q_k \cdot Z_t^j}{\|q_k\|_2 \cdot \left\|Z_t^j\right\|_2}) \\ &= \log(1 - \tilde{q}_j^{\top} \tilde{z}_k) + \log(1 - \tilde{q}_k^{\top} \tilde{z}_j) = \log(1 - \tilde{q}_j^{\top} \tilde{q}_k) + \log(1 - \tilde{q}_k^{\top} \tilde{q}_j)\end{aligned} \tag{6}$$

Then, the partial derivative can be deduced as:

$$\begin{aligned}\frac{\partial l(x_j, x_k)}{\partial q_j} &= -\frac{1}{1 - \tilde{q}_j^{\top} \tilde{q}_k} \cdot \frac{\partial(\tilde{q}_j^{\top} \tilde{q}_k)}{\partial q_j} - \frac{1}{1 - \tilde{q}_k^{\top} \tilde{q}_j} \cdot \frac{\partial(\tilde{q}_k^{\top} \tilde{q}_j)}{\partial q_j} \\ &= -\frac{1}{1 - \tilde{q}_j^{\top} \tilde{q}_k} \cdot \{\frac{1}{\left\|q_j\right\|_2}(\frac{q_k}{\|q_k\|_2} - \frac{q_j}{\left\|q_j\right\|_2} \frac{q_k}{\|q_k\|_2} \frac{q_j}{\left\|q_j\right\|_2}) \\ &\quad + \frac{1}{\|q_k\|_2}(\frac{q_j}{\left\|q_j\right\|_2} - \frac{q_k}{\|q_k\|_2} \frac{q_j}{\left\|q_j\right\|_2} \frac{q_k}{\|q_k\|_2})\}\end{aligned} \tag{7}$$

Given the stop-gradient strategy of Z for the proposed target network, we derive the magnitude of the gradient:

$$\begin{aligned}\left\|\frac{\partial l(x_j, x_k)}{\partial q_j}\right\|_2^2 &= \frac{1}{(1 - \tilde{q}_j^{\top} \tilde{q}_k)} \cdot \{\frac{1}{\left\|q_j\right\|_2^2}[1 - (\tilde{q}_j^{\top} \tilde{q}_k)^2] + \underbrace{\frac{1}{\|q_k\|_2^2}[1 - (\tilde{q}_k^{\top} \tilde{q}_j)^2]}_{=0} \\ &\quad + \underbrace{\frac{2}{\left\|q_j\right\|_2 \cdot \|q_k\|_2}(\tilde{q}_k^{\top} \tilde{q}_j - \tilde{q}_j^{\top} \tilde{q}_k - \tilde{q}_k^{\top} \tilde{q}_j + \tilde{q}_j^{\top} \tilde{q}_k)}_{=0}\} \\ &= (1 + \tilde{q}_j^{\top} \tilde{q}_k) \cdot \frac{1}{\left\|q_j\right\|_2^2}\end{aligned} \tag{8}$$

Let $m_j = 1/\left\|q_j\right\|_2^2$ and then we get a simplified form:

$$\left\|\frac{\partial l(x_j, x_k)}{\partial q_j}\right\|_2^2 = (1 + \tilde{q}_j^{\top} \tilde{q}_k) \cdot m_j. \tag{9}$$

According to Eq. (9), when $\tilde{q}_j$ and $\tilde{q}_k$ are in close proximity to one another, there is an increase in the gradient observed in the L2 norm. This gradient serves as a reliable indicator of the strength of pulling similar examples closer, implying that the gradient obtained from views of the same class is more substantial than that obtained from views of distinct classes. Especially, the gradient from the wrong sample pair $(x_j$, $x_f)$ is related to $\tilde{q}_j^{\top} \tilde{q}_f$, and $1+\tilde{q}_j^{\top} \tilde{q}_f \to 1$, which is evidently smaller than the value of $1+\tilde{q}_j^{\top} \tilde{q}_k$ for the correct sample pair (x_j, x_k), which is greater than 1. Therefore, the improved constraint function can suppress the gradient effect of the wrong label sample pair. This will not cause the model to overfit the clean samples because we connect a separate MLP as a classifier at the end of the projector, so as long as the gradient of the classification loss with respect to the parameters in the classifier is small on the clean samples, the model will not overfit them.

3.4. *Training with noisy labels*

As mentioned in the previous subsection, our training commences in the server-side. This choice is supported by our experimental findings, which suggest that commencing federated learning processes with a randomized initial model can be less effective than pre-training on the server's labeled dataset. The alternative approach not only results in faster convergence but also improves classification accuracy of the model.

3.4.1. *Server training*

The server is trained using noisy labels to generate the initial global model $\{\theta, \xi, \sigma\}$ through supervised learning and self-supervised contrastive representation learning. In the process of supervised learning, the available labeled dataset may be limited; therefore we employ image augmentation techniques to augment the size and quality of the dataset. In order to carry out a weak augmentation $A(\cdot)$, we resort to random image flipping and random image cropping. The augmented version of the initial input image x is denoted as $A(x)$. The supervised classification network on the server is composed of an encoder E_θ and a classifier C_σ, in which the parameters of the encoder and online network are shared. In cases where the label is devoid of errors or discrepancies, the preferred loss function to perform can be expressed as a form of cross-entropy loss:

$$\mathcal{L}_{ce} = -\sum_{i=1}^{M} y_i \cdot \log(\hat{y}_i), \tag{10}$$

where M indicates the number of samples, $\hat{y}_i$ is the prediction probability generated by the classifier for the labeled data x_i corresponding to the ground-truth label y_i. More specifically, we can express it as:

$$\hat{y}_i = \mathrm{softmax}(C_\sigma(G_\sigma(E_\sigma(x_i)))) \odot \mathrm{onehot}(y_i). \tag{11}$$

In cases of using the $\mathcal{L}_{ce}$ loss, the magnitude of the loss value tends to be higher when $\hat{y}_i$ approaches 0, which indicates that when labels are clean, the model leans towards focusing on samples likely to be incorrectly classified, yielding improved accuracy performance. However, in the presence of label noise, the effectiveness of the $\mathcal{L}_{ce}$ in distinguishing between samples that are likely to be misclassified and true error samples may be compromised, leading the model to over fit the noisy labels and impeding its ability to identify an optimal decision boundary. To address this issue, we make use of the generalized cross-entropy methodology as follows:

$$\mathcal{L}_{gce} = \sum_{i=1}^{M} y_i \cdot \frac{1 - (\hat{y}_i)^\gamma}{\gamma}, \tag{12}$$

where $\gamma \in (0, 1]$ is a non-negative coefficient. When $\gamma = 1$, the exponential term degenerates into the mean absolute error loss, and when $\gamma \to 0$, from L'Hôpital's rule, the exponential term degenerates into the cross-entropy loss. Eq. (12) benefits from both the mean absolute error loss and the cross-entropy loss. As the mean absolute error loss is a type of symmetric loss function, it guarantees impartial treatment of in the training process, hence establishing its robustness against noise.

Accordingly, we have devised comprehensive coping strategies for two different degrees of corrupted labels. When labels are clean, the total loss of the global model $\{\theta, \xi, \sigma\}$ on the server is given by:

$$\mathcal{L}_s = \mathcal{L}_{ce} + \lambda \mathcal{L}_{cr}, \tag{13}$$

where λ controls the strength of the contrastive regularization function. The collaborative effect of self-supervised learning and supervised learning prevents the model from memorizing erroneous labels and enhances model robustness. When the data labels held by the server contain noise, the following loss function can be used to deal with the challenge:

$$\mathcal{L}_s = \mathcal{L}_{gce} + \lambda \mathcal{L}_{cr}. \tag{14}$$

In certain instances, the utilization of the $\mathcal{L}_{gce}$ may be optional; provided the labels are precise, the $\mathcal{L}_{ce}$ remains the optimal approach. By leveraging this loss, we can effectively encourage representations from different categories to move away from each other and promote images of the same class to move closer, thereby improving model performance and combating noisy labels. After a certain round of server training, the parameters of the online model are split back to clients.

3.4.2. *Local model update*

K selected clients download the online network model parameters θ from the server to initialize their local contrastive networks. These clients then engage in unsupervised training, with the resulting loss being denoted as $\mathcal{L}_{cr}^k = \mathcal{L}_{cr}$ (cf. Eq. (4)). Following the client-k's local training process, we upload the client's local online network parameters W_k^θ, while the local target network parameters W_k^ξ can be discarded thereafter. This is due to the fact that in a long interval of time where one client is selected twice, the local target network parameter update schedule may not keep pace with the global online network, which could compromise the global network training process. Besides, incorporating this update approach offers several advantages, including the ability to learn knowledge from client's data without being affected by non-IID problems caused by class imbalance, conserving computing resources, and ensuring the protection of client data's security and privacy.

3.4.3. *Global model aggregation*

The server side is responsible for receiving the online network parameters uploaded by K selected clients. In the rth global round, the aggregation is performed through the utilization of the weighted averaging approach as the following calculation process:

$$\bar{W}^\theta(r+1) \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} \cdot W_k^\theta(r), \tag{15}$$

where n_k represents the number of samples owned by client k, while $n = \sum_{k=1}^{K} n_k$ denotes the total number of samples encompassed by selected clients in the system. We aggregate their parameters to get the global online network parameters for the r+1 round, denoted as $\bar{W}^\theta(r+1)$. Following this, the server then updates the global model by employing an integration of global online network parameters and other relevant model parameters that reside on the server side (cf. Section 3.4.1), which is then distributed to the newly sampled clients for the next round.

3.5. *Training algorithm*

A complete training pipeline of the proposed FedCR framework is shown in Algorithm 1, which is structured to achieve robust federated learning through an integration of client-side local update (lines 12–18) and server-side training (lines 19–26) and global aggregation (line 9). Initially, the global online network θ is pre-trained on the server's labeled dataset $\mathcal{D}_s$ for R_s rounds to establish a foundational model. Note that the local update methodology is completely consistent with the server-side training strategy. The federated learning iterative process spans from line 1 to line 11. Each communication round r involves sampling K clients from the total N available clients (line 4). The global model parameters are distributed to the selected clients, where each client performs local training independently using a contrastive learning objective over E_l epochs (lines 13–16). The locally updated model $W_k^\theta(r)$ is then uploaded back to the server (line 17). On the server, a global weighted-averaging aggregation operation is conducted to produce an updated global model, leveraging the received parameters and additional information such as ξ and σ for adjustment (lines 8–9).

Algorithm 1 The FedCR framework

```
Input: Global model $\{\theta,\xi,\sigma\}$; number of clients $N$ and dataset $D_i$ for
    client $i$; number of clients sampled $K$; global communication round
    $R_g$; the labeled dataset on the server $\mathcal{D}_s$; local training epochs $E_l$
Output: $W^{\theta,\xi,\sigma}$ from $R_g$-th round
 1: Server pre-trains the global model $\{\theta,\xi,\sigma\}$ on its labeled dataset $\mathcal{D}_s$
    for $R_s$ rounds.
 2: Server sends model $\{\theta,\xi,\sigma\}$ to all clients to initialize local models.
 3: for each federated learning round $r \in 1,2,\dots,R_g$ do
 4:     Randomly sample a subset $S^t$ of $K$ clients from $N$ clients
 5:     for each client $k \in S^t$ do
 6:         $W_k^{\theta}(r) \leftarrow$ LocalUpdate$(k, \bar{W}^{\theta}(r))$
 7:     end for
 8:     Perform global models aggregation to get $\bar{W}^{\theta}(r+1)$ via Eq. (15)
 9:     $\bar{W}^{\theta}(r+1) \leftarrow$ ServerAggregation$(\{W_k^{\theta}(r)\}_{k\in S^t})$
10:     $\{\theta,\xi,\sigma\} \leftarrow$ ServerTraining $(\bar{W}^{\theta}(r+1), W^{\xi}, W^{\sigma})$
11: end for
12: function LocalUpdate($k$, $W^{\theta}$)
13:     $W_k^{\theta} \leftarrow W^{\theta}, W_k^{\xi} \leftarrow W^{\theta}$          ▷ Initialize local model
14:     for each local epoch $i = \{1,2,\dots,E_l\}$ do
15:         Perform local contrastive learning as Eq. (4)
16:     end for
17:     return $W_k^{\theta}$    ▷ Upload the updated online network parameters
18: end function
19: function ServerTraining($\bar{W}^{\theta}$, $W^{\xi}$, $W^{\sigma}$)
20:     for each local epoch $i = 1,2,\dots,E_s$ do
21:         $\mathcal{B} \leftarrow$ (split $\mathcal{D}_s$ into batches of size $B$)
22:         for each batch $b \in \mathcal{B}$ do
23:             Update global model $\{\theta,\xi,\sigma\}$ by minimizing $\mathcal{L}_s$ as Eq.
    (14)
24:         end for
25:     end for
26: end function
```

Subsequently, server-side training (lines 19–26) is executed, where the global model is fine-tuned using mini-batches of the server dataset $\mathcal{D}_s$, optimizing the loss function $\mathcal{L}_s$ over E_s epochs. This iterative process of local training as well as global aggregation and updating ensures that the model learns collaboratively from distributed datasets while maintaining client data privacy. The framework is designed to cope with the issues of federated semi-supervised learning with noisy labels by incorporating both local and global optimization steps, promoting the learning of unlabeled data across different clients.

4. Experiments

In this section, we conduct experiments on three benchmark datasets on the performance of our proposed FedCR in IID and non-IID settings, as well as its robustness to different types and levels of noise. In addition, we perform ablation experiments to analyze the effectiveness of important components of our model. At the end of this section, we discuss and analyze the limitations of the method.

4.1. Datasets

We conduct experiments for the image classification task on three artificially corrupted datasets SVHN [80], CIFAR-10 [81] and CIFAR-100 [81]. To realistically mimic noisy datasets, we consider two different types of synthetic noise with various noise levels in those three datasets, including symmetric noise and asymmetric noise. For symmetric noise, each label has the same probability of being incorrectly labeled as any other classes. For asymmetric noise, different classes of labels have different probability of being incorrectly labeled. (1) SVHN (Street View House Numbers) is a real-world image dataset from Google street view images, consisting of cropped images of house numbers taken from street level images (10 classes). We choose 5000 labeled images for training. (2) CIFAR-10: We randomly select 5000 samples from 60,000 samples as labeled data, where each class contains 500 randomly extracted samples (the total number of class is 10). (3) CIFAR-100: We also choose 5000 labeled samples from 60,000 samples, where each class contains 50 randomly extracted instances. For all datasets, the generation for symmetric label noise levels $r \in \{20\%, 40\%, 80\%\}$. For asymmetric label noise of CIFAR-10, we follow the setting in [82] by mapping TRUCK → AUTOMOBILE, BIRD → AIRPLANE, DEER → HORSE, and CAT ↔ DOG with probability 40%. To generate asymmetric label noise for the other datasets, we randomly sample 40% of the data and flip their labels to the next class.

4.2. Baselines

To demonstrate the effectiveness and robustness of our proposed FedCR, we evaluate FedCR under different settings such as unlabeled data ratios, data distributions, and types of noise labels. Specifically, we consider the following baselines or state-of-the-art federated semi-supervised learning methods in our experiments, including: (1) Server-SL: A supervised learning model trained exclusively on labeled data, without the use of any unlabeled data. (2) FedAvg-FixMatch: Naive combination of FedAvg [1] and FixMatch [83]. (3) FedProx-FixMatch: Naive combination of FedProx [84] and FixMatch. (4) FedAvg-UDA: Naive combinations of FedAvg with UDA [85]. (5) FedProx-UDA: Naive combinations of FedProx with UDA. (6) FedMatch: A FedSSL algorithm proposed in [7] with inter-client consistency loss and parameter decomposition. (7) Fedcon: A contrastive FedSSL method using a two top-layer structure to solve the decoupling problem of labeled and unlabeled data [41]. (8) SSFL: A federated semi-supervised learning algorithm with group normalization and consistency regularization [8]. (9) SemiFL [42]: A semi-supervised federated learning framework that alternates between training labeled the server and unlabeled clients. (10) RSCFed: A FedSSL algorithm proposed in [34] with random sampling consensus. (11) FedAnchor: A FedSSL method with label contrastive loss based on a cosine similarity measure, used for training labeled anchor data on servers [44]. (12) $(FL)^2$ [45] is a training method for unlabeled clients using sharpness-aware consistency regularization. In order to compare our method with existing FedSSL methods, we follow the standard FL setting originally used in FedAvg and widely adopted by most works.

4.3. Experimental details

In the contrastive representation module, we utilize standard data augmentation techniques to generate two distinct random views, which encompass cropping, flipping, color jitter, and gray-scale adjustments. The input samples are randomly cropped with a ratio ranging from 0.2 to 1.0, and color jitter is applied to adjust brightness, contrast, saturation, and hue, with intensities set to 0.4, 0.4, 0.4, and 0.1, respectively. The probability of applying these augmentations is set at 0.8. Additionally, random gray-scale is applied to the samples with a probability of 0.2. Our approach utilizes pre-training PreAct ResNet-18 [86] as the feature extraction backbone and encoder. Compared with the ResNet, the PreAct ResNet rearranges the order of the Conv-BN-ReLU layers, creating a path from the first ResNet block to the last ResNet block without the intermediate non-linear transformation of ReLU. This modification has been shown to improve the accuracy of the model. In addition, we use a multi-layer perceptron (MLP) as a projector, which has 4096 hidden dimensions and output 2048-dimensional embedding. The predictor also adopts an MLP with the same architecture. Fig. 4 shows the data processing flow of the encoder and the detailed structure of the internal network layers.

We reproduce these methods in our framework based on public codes and trained our model using the PyTorch framework toolbox in

Table 1
Parameters chosen in the experiments.

Symbol	Value	Description
R_g	200	Global training round
R_l	5	Local epochs
R_s	700	Server pre-training round
N	100/90	Total clients
K	5	Active clients
r	0/0.2/0.4/0.8	Noise label ratio
B	64	Batch size
γ	0.6	L_{gce} parameter
η	0.8	L_{cr} parameter
λ	50	The strength of the contrastive regularization

Python on a computing server with Intel (R) Xeon (R) @ 2.20 GHz CPU and NVIDIA GeForce GTX 3090 GPU. The optimizer is set with the Adam optimization [87], where the learning rate is 0.003 and the decay rate is 0.99. To prevent possible model collapsing, the minimum batch size of the contrastive representation network is generally set to 128. However, given that each client has only 500 or fewer samples, we set the batch size to 64 to improve the learning of unlabeled data in the client-side. For CIFAR-10 and SVHN datasets, we set up 450 unlabeled instances for 100 clients, with randomly extracted 45 unlabeled instances for each class in the IID scenario. In the non-IID scenario, we generate random class distributions. For CIFAR-100 dataset, we extract 500 unlabeled instances for each client, with randomly 5 instances for each class in the IID scenario. For non-IID tasks, the class distribution of 500 instances in each client is random. After non-IID data partitioning, the class and sample size of each client are different. Our method involves pre-training on the server side first, followed by 200 rounds of federated learning. For each federated learning round, we randomly select 5 clients to participate in the training process. Table 1 provides further details on the parameters used in our experiment.

4.4. Experimental results

4.4.1. Efficiency evaluation for the IID and non-IID tasks

Table 2 showcases the test accuracy results on CIFAR-10, CIFAR-100, and SVHN datasets achieved by various methods in both the IID and non-IID scenarios. Among these, the Server-SL approach has been trained on the server-side using a limited quantity of labeled data, which is the lower-bound baseline of all comparison methods. Besides, we incorporate another four baseline methods: FedAvg-FixMatch, FedProx-FixMatch, FedAvg-UDA, and FedProx-UDA. These methods represent different combinations of classical semi-supervised learning approaches (FixMatch and UDA) with federated learning methods (FedAvg and FedProx). We employ a fixed confidence score threshold of 0.95 for FixMatch and 0.8 for UDA. Furthermore, we compare our FedCR approach with state-of-the-art federated semi-supervised learning methods, including FedMatch, RSCFed, SSFL, Fedcon, SemiFL, FedAnchor, and $(FL)^2$ in the benchmark results. We test the image classification accuracy of these methods on different datasets at both IID and non-IID conditions. Fig. 5 shows the performance comparison with error bar between our method and the latest method on three different datasets, under IID and non-IID data distributions.

From the results, we can find that FedCR practically outperforms competitors in terms of measurement performance in all cases. FedCR has surpassed the average accuracy of Server-SL, FedMatch, SemiFL, and $(FL)^2$ in all datasets by 30.85%, 28.81%, 7.79%, and 3.79% respectively. In the presence of noisy labels, these methods tend to suffer from a degradation in performance, as they may rely on the quality of the pseudo-labels generated from the unlabeled data. When these labels are incorrect, they can mislead the model during training, especially in federated settings where the model updates from various clients are aggregated. In contrast, FedCR incorporates noise-robust loss functions that can reduce the impact of erroneous labels, enabling it to maintain higher accuracy even in noisy settings. While the accuracy of our model in the non-IID setting is marginally lower than its performance in the IID setting in CIFAR-100 dataset, the FedCR still provides notable advantages over all other considered methods. This is due to data heterogeneity, which may limit the number of classes in the client's local dataset. For instance, the client's local dataset might only contain two categories, making it more difficult to identify the correct category for a given image. While methods like Fedcon and SemiFL struggle to generalize across heterogeneous data, especially when the clients' local datasets are sparse or highly imbalanced. FedCR is designed to more effectively handle non-IID data by incorporating strategies that stabilize the client model updates, ensuring that the global model is more robust to data heterogeneity. Moreover, the classification accuracies of the test models on the CIFAR-100 dataset are found to be lower compared to their performance on the CIFAR-10 dataset due to the larger number of classes and challenging scenes on the CIFAR-100 dataset and the relatively smaller number of labeled examples available for training. The increase in the number of classes leads to greater label ambiguity, especially when a significant portion of the labels are noisy or when the client-side data is highly unbalanced. Other methods struggle in such situations because their reliance on inaccurate labels exacerbates the class imbalance and reduces the overall classification accuracy. FedCR's ability to effectively filter noisy labels and leverage both labeled and unlabeled data in a robust manner helps it achieve better accuracy, even under the difficult conditions posed by CIFAR-100. For the simpler street view house number dataset SVHN, our results also show a high advantage. By examining the accuracy across different data distributions, it is clear that these algorithms do not always exhibit a straightforward positive correlation with data heterogeneity levels. For instance, in the more challenging CIFAR-100 dataset, $(FL)^2$ and FedAnchor demonstrate impressive accuracy increases of 2.11% and 1.81% compared to IID under non-IID conditions. Additionally, in the SVHN dataset, FedCR achieves a slight decrease of 5.98% with non-IID distribution, while still maintaining the highest predictive performance. The slight drop in accuracy under non-IID conditions can be attributed to the increased difficulty of classifying images when data is more spread out across clients. However, unlike other methods that suffer much more in these conditions, FedCR remains relatively stable, showing its robustness across different scenarios.

Simultaneously, it is worth noting that the test accuracies of the four naive combinations methods FedAvg-FixMatch, FedProx-FixMatch, FedAvg-UDA and FedProx-UDA are almost equivalent to or inferior to the performance of Server-SL, especially in the presence of noisy labels. The reasons for their subpar performance can be explained as follows: First, these methods simply combine federated learning with semi-supervised learning (SSL) algorithms without considering the impact of noisy labels on the model's performance. In semi-supervised learning, the model relies on pseudo-labels to train on the unlabeled data. However, when client-side data distributions are heterogeneous or when the labels are noisy, the pseudo-labels generated by the SSL algorithms are often inaccurate or even completely wrong. These erroneous labels lead to bias during local training on the client side, which is then propagated and aggregated in later rounds. As a result, the global model's performance deteriorates due to the accumulation of these inaccuracies. Second, FL algorithms like FedAvg and FedProx assume that the model updates from each client are reliable. However, in the presence of noisy labels, this assumption no longer holds true. The local models, influenced by incorrect labels, generate biased updates that drift away from the true data distribution. When these updates are aggregated in a federated setting, the resulting global model is further skewed, which affects the model's generalization ability. Additionally, SSL algorithms like FixMatch and UDA are inherently sensitive to label noise. These methods typically rely on confidence thresholds to select pseudo-labels, assuming that high-confidence pseudo-labels are correct. However, in the presence of noisy labels, even high-confidence pseudo-labels can be incorrect. This leads to the model being misled during training, as it continuously learns from inaccurate pseudo-labels, resulting in poor performance.

Table 2
Average accuracy results on CIFAR-10, SVHN, and CIFAR-100 datasets under the IID setting and non-IID data partition. Note that Server-SL method represents only the results of training on the server.

Method	CIFAR-10		CIFAR-100		SVHN	
	IID	Non-IID	IID	Non-IID	IID	Non-IID
Server-SL	54.21$_{\pm0.14}$	–	14.21$_{\pm0.24}$	–	59.03$_{\pm0.53}$	–
FedAvg [1]-FixMatch [83]	54.61$_{\pm0.21}$	53.84$_{\pm0.34}$	13.50$_{\pm0.22}$	13.35$_{\pm0.17}$	64.57$_{\pm0.20}$	62.24$_{\pm0.14}$
FedProx [84]-FixMatch [83]	53.26$_{\pm0.20}$	52.13$_{\pm0.30}$	13.84$_{\pm0.15}$	13.71$_{\pm0.09}$	55.38$_{\pm0.46}$	49.38$_{\pm0.20}$
FedAvg [1]-UDA [85]	46.66$_{\pm0.54}$	45.37$_{\pm0.19}$	11.56$_{\pm0.13}$	11.41$_{\pm0.14}$	57.49$_{\pm0.29}$	52.64$_{\pm0.57}$
FedProx [84]-UDA [85]	53.49$_{\pm0.53}$	52.98$_{\pm0.12}$	14.53$_{\pm0.14}$	13.57$_{\pm0.07}$	74.94$_{\pm0.28}$	70.63$_{\pm0.91}$
FedMatch [7]	56.25$_{\pm0.30}$	54.40$_{\pm0.23}$	21.80$_{\pm0.12}$	21.04$_{\pm0.11}$	68.33$_{\pm0.68}$	64.12$_{\pm1.20}$
RSCFed [34]	55.05$_{\pm0.33}$	53.33$_{\pm0.17}$	13.61$_{\pm0.09}$	11.97$_{\pm0.04}$	75.25$_{\pm0.83}$	79.32$_{\pm0.94}$
SSFL [8]	47.62$_{\pm0.10}$	46.32$_{\pm0.11}$	13.21$_{\pm0.08}$	13.13$_{\pm0.09}$	82.05$_{\pm0.57}$	75.49$_{\pm0.28}$
Fedcon [41]	57.28$_{\pm0.41}$	51.85$_{\pm0.53}$	21.43$_{\pm0.14}$	17.51$_{\pm0.45}$	72.04$_{\pm0.56}$	69.29$_{\pm0.52}$
SemiFL [42]	77.27$_{\pm0.30}$	71.27$_{\pm0.26}$	41.31$_{\pm1.23}$	32.04$_{\pm1.41}$	75.30$_{\pm0.18}$	71.35$_{\pm0.83}$
FedAnchor [44]	78.74$_{\pm0.64}$	72.27$_{\pm0.63}$	38.83$_{\pm0.62}$	40.64$_{\pm0.21}$	86.48$_{\pm0.27}$	85.38$_{\pm1.04}$
$(FL)^2$ [45]	81.27$_{\pm1.50}$	78.63$_{\pm2.48}$	45.63$_{\pm1.74}$	47.74$_{\pm2.02}$	92.48$_{\pm2.19}$	87.10$_{\pm1.46}$
FedCR	**85.06**$_{\pm\mathbf{0.07}}$	**80.15**$_{\pm\mathbf{0.16}}$	**47.89**$_{\pm\mathbf{0.33}}$	**49.37**$_{\pm\mathbf{0.20}}$	**94.29**$_{\pm\mathbf{0.36}}$	**88.31**$_{\pm\mathbf{0.39}}$

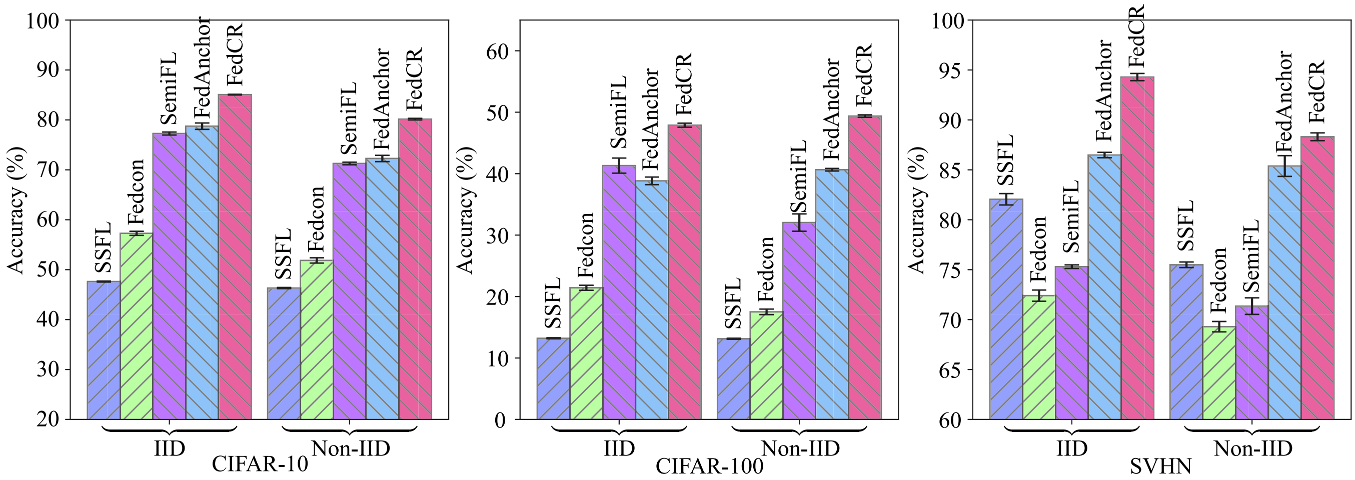


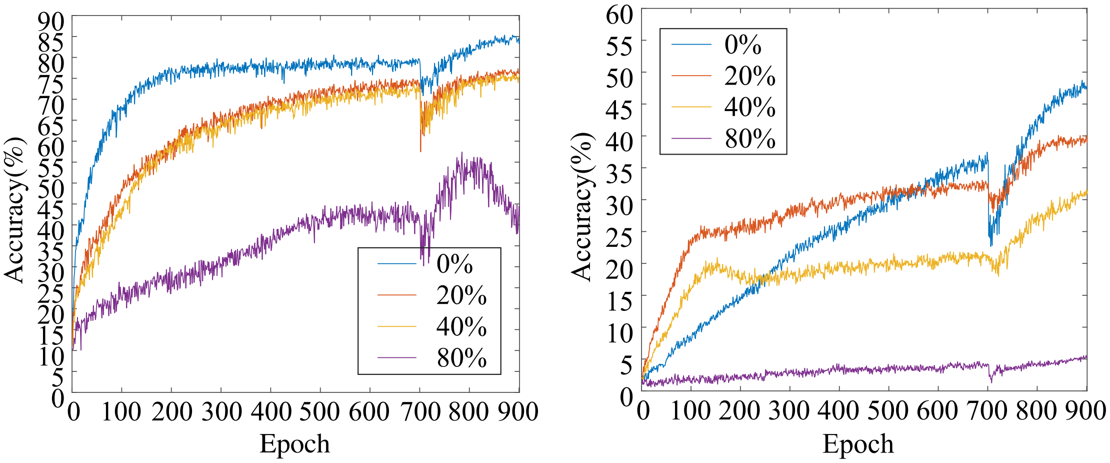
Fig. 5. Test accuracy (%) on different number of local clients K selected in each federated semi-supervised learning round.

Table 3
Test accuarcy on CIFAR-10 and CIFAR-100 with different noise types and noise levels.

Method	CIFAR-10						CIFAR-100					
	Symmetrical			Asymmetric			Symmetrical			Asymmetric		
	20%	40%	80%	20%	40%	80%	20%	40%	80%	20%	40%	80%
Server-SL	42.01$_{\pm0.11}$	33.33$_{\pm0.05}$	13.92$_{\pm0.08}$	52.04$_{\pm0.46}$	45.88$_{\pm0.09}$	40.36$_{\pm0.82}$	8.14$_{\pm0.14}$	5.53$_{\pm0.11}$	4.16$_{\pm0.02}$	9.02$_{\pm0.48}$	5.73$_{\pm0.07}$	5.28$_{\pm0.07}$
FedAvg-FixMatch	39.49$_{\pm0.25}$	31.17$_{\pm0.18}$	10.05$_{\pm0.10}$	54.28$_{\pm0.28}$	50.68$_{\pm0.36}$	47.03$_{\pm0.48}$	11.64$_{\pm0.08}$	5.86$_{\pm0.05}$	4.66$_{\pm0.04}$	9.47$_{\pm0.58}$	5.87$_{\pm0.04}$	5.39$_{\pm0.23}$
FedProx-FixMatch	44.68$_{\pm0.24}$	28.57$_{\pm0.19}$	10.00$_{\pm0.01}$	50.83$_{\pm0.16}$	44.86$_{\pm0.19}$	39.57$_{\pm0.31}$	8.44$_{\pm0.12}$	5.78$_{\pm0.08}$	4.78$_{\pm0.03}$	8.59$_{\pm0.41}$	5.61$_{\pm0.02}$	4.95$_{\pm0.17}$
FedAvg-UDA	41.68$_{\pm0.42}$	27.88$_{\pm0.38}$	12.12$_{\pm0.51}$	49.28$_{\pm0.52}$	44.06$_{\pm0.44}$	38.28$_{\pm0.13}$	9.78$_{\pm0.07}$	4.76$_{\pm0.03}$	2.02$_{\pm0.04}$	10.42$_{\pm0.52}$	6.41$_{\pm0.07}$	6.01$_{\pm0.18}$
FedProx-UDA	44.08$_{\pm0.20}$	35.20$_{\pm0.14}$	14.10$_{\pm0.12}$	52.59$_{\pm0.52}$	44.31$_{\pm0.19}$	42.48$_{\pm0.28}$	9.69$_{\pm0.12}$	5.24$_{\pm0.10}$	1.89$_{\pm0.05}$	12.43$_{\pm0.34}$	6.44$_{\pm0.03}$	7.32$_{\pm0.54}$
FedMatch	46.37$_{\pm0.23}$	38.11$_{\pm0.29}$	17.50$_{\pm0.21}$	56.28$_{\pm0.41}$	47.80$_{\pm0.20}$	45.91$_{\pm0.42}$	18.73$_{\pm0.07}$	8.30$_{\pm0.07}$	1.42$_{\pm0.08}$	16.30$_{\pm1.03}$	10.05$_{\pm0.15}$	8.23$_{\pm0.41}$
RSCFed	46.24$_{\pm0.21}$	32.99$_{\pm0.23}$	15.60$_{\pm0.17}$	47.28$_{\pm0.18}$	41.51$_{\pm0.22}$	43.40$_{\pm0.57}$	9.32$_{\pm0.05}$	6.08$_{\pm0.05}$	5.13$_{\pm0.10}$	11.38$_{\pm1.34}$	6.39$_{\pm0.03}$	7.39$_{\pm0.57}$
SSFL	40.33$_{\pm0.08}$	27.01$_{\pm0.10}$	10.01$_{\pm0.10}$	49.20$_{\pm0.19}$	41.37$_{\pm0.19}$	46.29$_{\pm1.28}$	9.21$_{\pm0.08}$	5.89$_{\pm0.04}$	4.23$_{\pm0.03}$	9.42$_{\pm0.42}$	6.41$_{\pm0.08}$	7.57$_{\pm1.32}$
Fedcon	50.27$_{\pm0.32}$	49.26$_{\pm0.15}$	37.18$_{\pm0.76}$	55.57$_{\pm0.84}$	43.20$_{\pm0.93}$	46.02$_{\pm0.54}$	12.35$_{\pm0.53}$	7.89$_{\pm2.01}$	5.04$_{\pm0.63}$	25.83$_{\pm0.76}$	21.44$_{\pm0.62}$	9.32$_{\pm0.32}$
SemiFL	68.16$_{\pm0.53}$	63.07$_{\pm0.82}$	45.15$_{\pm0.34}$	60.17$_{\pm0.46}$	58.25$_{\pm0.24}$	56.38$_{\pm0.62}$	27.37$_{\pm0.81}$	12.54$_{\pm0.59}$	4.49$_{\pm0.73}$	36.04$_{\pm1.04}$	17.39$_{\pm1.39}$	10.02$_{\pm0.42}$
FedAnchor	64.26$_{\pm0.26}$	63.28$_{\pm0.82}$	49.15$_{\pm0.82}$	67.25$_{\pm0.34}$	71.28$_{\pm0.91}$	50.82$_{\pm0.42}$	30.43$_{\pm0.53}$	18.38$_{\pm0.71}$	5.23$_{\pm0.14}$	32.47$_{\pm0.55}$	28.42$_{\pm0.35}$	8.32$_{\pm0.70}$
$(FL)^2$	72.82$_{\pm0.63}$	70.26$_{\pm0.72}$	49.01$_{\pm0.31}$	**75.27**$_{\pm\mathbf{1.02}}$	65.36$_{\pm0.62}$	54.28$_{\pm0.64}$	36.35$_{\pm0.82}$	29.35$_{\pm0.54}$	5.40$_{\pm0.93}$	36.04$_{\pm0.72}$	**31.23**$_{\pm\mathbf{0.44}}$	10.93$_{\pm0.83}$
FedCR	**76.78**$_{\pm\mathbf{0.91}}$	**74.63**$_{\pm\mathbf{0.75}}$	**50.11**$_{\pm\mathbf{0.64}}$	75.21$_{\pm0.21}$	**72.49**$_{\pm\mathbf{0.50}}$	**58.24**$_{\pm\mathbf{0.10}}$	**39.89**$_{\pm\mathbf{0.26}}$	**31.10**$_{\pm\mathbf{0.12}}$	**5.48**$_{\pm\mathbf{0.10}}$	**37.35**$_{\pm\mathbf{0.13}}$	30.42$_{\pm0.08}$	**11.20**$_{\pm\mathbf{0.32}}$

4.4.2. Efficiency evaluation with different level of noisy labels

The experimental results on the CIFAR-10 and CIFAR-100 datasets under different label noise settings are presented in Table 3. To examine the impact of varying categories and noise levels on our model's performance, we apply contrastive learning and the generalized cross-entropy loss function in FedCR during server-side supervised learning. This design choice specifically addresses the detrimental effects of noisy labels by reducing the influence of incorrect labels on the training process, making it particularly effective in scenarios where traditional loss functions such as cross-entropy tend to amplify the impact of noise. For each dataset, we design two distinct label corruption schemes and employ three different noise addition ratios: 20%, 40%, 80%. Our method consistently yields superior performance compared to other methods, across almost all noise settings tested, demonstrating exceptional robustness. As the noise level on the CIFAR-10 dataset increases to 80%, it is obvious that other methods' models are considerably affected by a large amount of noise, making it difficult to accurately classify the test samples, resulting in significant performance degradation. These methods, which often lack robust noise-handling mechanisms, are vulnerable to the inherent label noise, causing local models to learn incorrect patterns. In contrast, FedCR leverages the GCE loss function and contrastive learning strategies that minimize the



(a) CIFAR-10

(b) CIFAR-100

Fig. 6. The performance changes of the model affected by different noise level on two datasets. (a) Test Accuracy(%) on the CIFAR-10 dataset with symmetric label noise. (b) Test Accuracy(%) on the CIFAR-100 dataset with symmetric label noise.

effect of noisy labels on the global model. This significantly improves the model's resilience, resulting in over 50% better accuracy than Server-SL in extreme noise scenarios. In comparison to more advanced methods like Fedcon, SemiFL, FedAnchor, and $(FL)^2$, our method demonstrates clear advantages. On the challenging CIFAR-100 dataset, which involves a more complex classification task due to a larger number of classes and more intricate image features, our approach achieves optimal results even with high proportions of noisy labels. This highlights the effectiveness of FedCR in dealing with challenging, noise-prone conditions, minimizing the impact of noise while maximizing the utility of the available data.

Additionally, we investigate the effect of varying levels of symmetric noise on the performance of the model across CIFAR-10 and CIFAR-100, both of which pose different degrees of classification difficulty. Fig. 6 illustrates the impact of label noise on model performance, where the initial 700 epochs in the curve showcase pre-training phase on the server-side. During this phase, the model achieves a high degree of accuracy under supervised training, but the transition to federated learning inevitably causes a drop in performance. This decline is primarily due to client-side data heterogeneity, where inconsistent data distributions across clients lead to conflicting local updates that degrade the global model's performance in the early rounds. However, FedCR quickly compensates for this initial drop by leveraging adaptive update strategy and aggregation, enabling the global model to stabilize and gradually enhance its performance in subsequent rounds. As evidenced by the data in Fig. 6, an increase in the ratio of label noise leads to a clear decline in the model's performance. Furthermore, the evaluation results on the CIFAR-100 dataset reveal poor classification performance due to a large number of total classes, and the severity of noise interference impacts classification accuracy.

4.5. Ablation study

In addition to our main results, we further conduct ablation studies on the CIFAR-10 and SVHN datasets to evaluate major components within FedCR.

4.5.1. Effects of the learning objectives

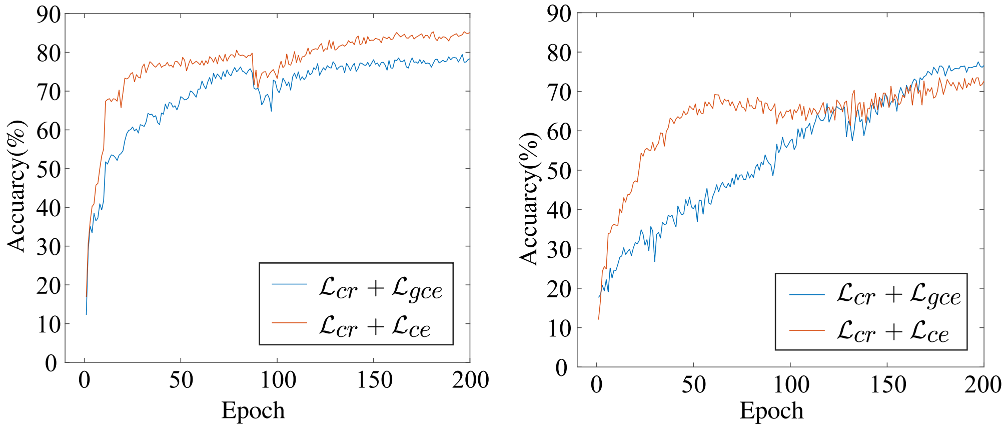
Firstly, we assess how the introduced learning objectives affect the effectiveness of our method. The results are summarized in Table 4. Specifically, we experimented with different variants of FedCR by removing certain components and compared their performance to that of the original method with 40% noise labels. Our analysis indicates that using generalized cross-entropy loss facilitates fair handling of labels during the training process. This approach bolsters the model's robustness to noise and stabilizes decision boundaries, ultimately improving training performance. However, relying solely on $\mathcal{L}_{gce}$ loss may lead to significant divergence in local objectives, hindering collaborative knowledge learning. As shown in Table 4, the model experiences benefits from contrastive regularization of 3.21%, and 2.67% across both datasets. The data from the second and third rows of the table highlight the importance of $\mathcal{L}_{gce}$ and $\mathcal{L}_{cr}$ for enhancing model performance. Specifically, removing $\mathcal{L}_{gce}$ results in performance declines of 2.27% and 1.75%, respectively, in the final accuracy metric. In summary, the removal of any loss term from these two components leads to varying degrees of decline across the performance indicator. These declines underscore the significance of the constraint mechanisms integrated into these two items for ensuring the final model's performance. Overall, the model performs optimally when both the proposed contrastive representation regularization and enhanced cross-entropy loss are employed.

Table 4
Ablation study of loss functions and EMA strategy on CIFAR-10 and SVHN datasets.

Method	CIFAR-10 (%)	SVHN (%)
FedCR	$74.63_{\pm0.75}$	$89.10_{\pm0.12}$
(w/o) $\mathcal{L}_{gce}$	$72.36_{\pm0.43}$ (↓2.27)	$87.35_{\pm0.21}$ (↓1.75)
(w/o) $\mathcal{L}_{cr}$	$71.42_{\pm0.37}$ (↓3.21)	$86.43_{\pm0.45}$ (↓2.67)
(w/o) EMA	$72.04_{\pm0.25}$ (↓2.59)	$83.53_{\pm0.74}$ (↓5.57)

4.5.2. Effects of the contrastive regularization function

In FedCR, the contrastive regularization function plays a crucial role on supervised and unsupervised training of both the server and client models. We conduct several experiments to demonstrate the efficiency of this function, particularly by comparing it with the contrastive loss function used in the BYOL model denoted as $\mathcal{L}_{mse}$ under IID and non-IID conditions. The comparison results presented in Table 5 reveal that the contrastive regularization function improves the model's prediction performance by almost 8% compared to the traditional contrastive loss function. This confirms that compared to the $\mathcal{L}_{mse}$ loss, the proposed contrastive regularization function $\mathcal{L}_{cr}$ is capable of improving the learning of category features by enhancing gradient updates for correct sample pairs while minimizing updates for mislabeled sample pairs. By using predicted probability distributions to filter out noisy labels, we can bring images that genuinely belong to the same category closer together, while pushing images from different categories apart. This outcome aligns with our theoretical analysis.



(a) r=0%

(b) r=20%

Fig. 7. Performance of FedCR with different supervised learning loss functions under different noise level. In cases where the label noise is absent, the L_{ce} is considered optimal, whereas the use of the L_{gce} is recommended when label noise is present. (a) Test accuracy when $r = 0\%$. (b) Test accuracy when $r = 20\%$.

Table 5
Ablation study on our method (FedCR) about comparing contrastive regularization function with other methods under IID and non-IID settings.

Method	CIFAR-10		SVHN	
	IID	Non-IID	IID	Non-IID
$\mathcal{L}_{mse}$	$77.63_{\pm 0.05}$	$76.62_{\pm 0.07}$	$90.36_{\pm 0.24}$	$84.23_{\pm 0.16}$
$\mathcal{L}_{cr}$	$85.06_{\pm 0.07}$	$80.05_{\pm 0.16}$	$94.29_{\pm 0.36}$	$88.31_{\pm 0.39}$

Table 6
Ablation study on our method (FedCR) about comparing different supervised loss function under different levels of label noise.

Method	CIFAR-10		CIFAR-100	
	0%	20%	0%	20%
$\mathcal{L}_{ce}$	$85.06_{\pm 0.07}$	$71.86_{\pm 0.31}$	$38.92_{\pm 0.34}$	$36.45_{\pm 0.17}$
$\mathcal{L}_{gce}$	$78.07_{\pm 0.33}$	$76.78_{\pm 0.91}$	$40.61_{\pm 0.13}$	$39.89_{\pm 0.26}$

Table 7
The result of different model update methods in IID and non-IID tasks.

Method	CIFAR-10	
	IID	Non-IID
Whole model	$84.58_{\pm 0.33}$	$84.41_{\pm 0.20}$
FedEMA	$81.59_{\pm 0.21}$	$81.01_{\pm 0.19}$
FedCR	$85.06_{\pm 0.07}$	$80.15_{\pm 0.16}$

4.5.3. *Effects of the generalized cross-entropy loss function*

In order to assess the impact of the generalized cross-entropy loss function on label correction in the presence of noisy labeled data, we conduct an experiment utilizing the cross-entropy loss function as a comparison. As shown in Fig. 7(a), we observe that the model's precision is higher when using the cross-entropy loss function in the absence of label noise, demonstrating the effectiveness and superiority of the cross-entropy loss optimization model when the labels are clean.

In Fig. 7(b), we present the training results in the presence of label noise ($r = 20\%$). Employing the cross-entropy loss function can make model converge quickly and towards stability, however, it has potential to cause over-fitting in case of inaccurate labels. Despite inheriting the disadvantage of slow convergence from the symmetric loss function, the generalized cross-entropy loss function improves the model's resistance to noisy labels. As shown in Table 6, during the 200-epoch training period, the model achieves a 4.92% performance improvement, proving its ability to train a robust model under noisy labels.

4.5.4. *Effects of the model update method*

We evaluate our model update approach in a federated learning environment and compare it with two other model update methods in Table 7. The first approach is the classic federated approach, which involves exchanging the entire contrastive representation network, including the online network and target network between the server and clients. The second approach is FedEMA [88], in which the local model of clients adaptively using EMA of the global model while the server and client exchange online network parameters. The attenuation rate is dynamically measured by the model's divergence. We compare the three methods under IID and non-IID settings, and the results in Table 7 show that our approach performs the best in the IID environment. While the performance in the non-IID setting is slightly lower compared to the other two methods, this outcome is anticipated due to the uneven distribution of client data in this scenario. Additionally, since the client's target network parameters are discarded in each federated learning round, some local knowledge may not be adequately preserved and learned by the global model, resulting in a decrease in the accuracy of model training. However, an important advantage of our method is that it safeguards data privacy by not uploading target network parameters, while also decreasing computational demands and minimizing expenses related to clients' local storage.

4.5.5. *Effects of the number of clients*

The primary hyper-parameter of our model is the count of local clients, denoted as K. An analysis is conducted on the number of clients selected for each round, and the results in Fig. 8 show that there is only a slight variation in accuracy of the model with varying K values. The model performance is usually better when K is set to 5 or 30. However, considering the training efficiency of the federated learning system, we choose to select a smaller value based on experience and set $K = 5$. One possible explanation for this finding is that in our method, the focus of client execution is unsupervised training, which requires more training epochs. Including too many client aggregations in one round does not promote the final model performance. This proves on another level the high tolerance of our method for client engagement rate.

4.5.6. *Effects of the threshold η and γ*

In our model, the threshold η in contrastive regularization loss is also the main hyperparameter we study. In FedCR, we use $\mathcal{L}_{cr}$ for model training and optimization. The η in $\mathcal{L}_{cr}$ indicates the effectiveness of the Indicator function, and different threshold settings can impact the model's training performance. We evaluated the training accuracy of FedCR at various η values, specifically at levels {0.2, 0.4, 0.6, 0.8, 1}.

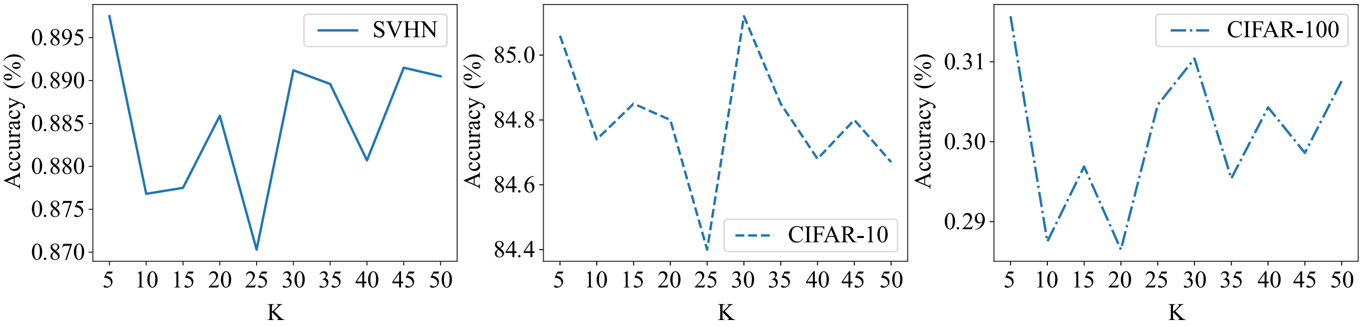


Fig. 8. Test accuracy (%) on different number of local clients K selected in each federated semi-supervised learning round.

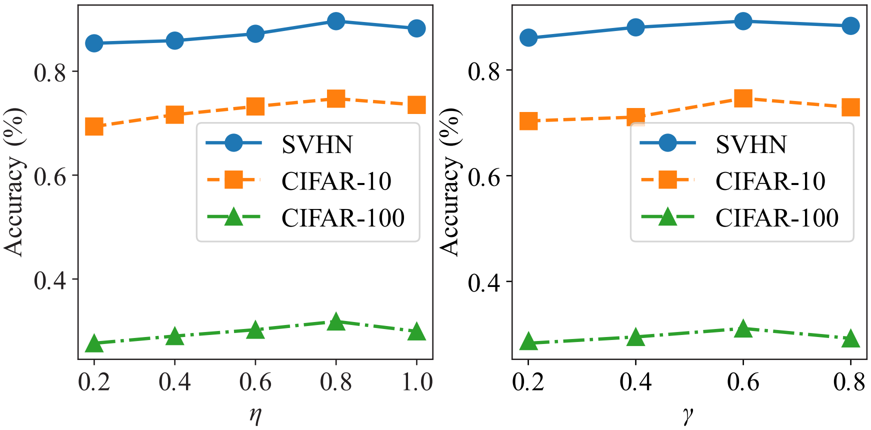


Fig. 9. Test accuracy (%) on different threshold η and γ.

From the left curve of Fig. 9, we observe that as the η level increases, FedCR exhibits a nearly consistent trend in accuracy performance across different datasets, initially increasing and then decreasing, with the best performance at an η value of 0.8. We also conduct sensitivity analysis experiments on the γ parameter in the generalized cross entropy loss. As it is a non-negative number between $(0, 1]$, we limit its range to {0.2, 0.4, 0.6, 0.8}. From the experimental results in the right part of Fig. 9, we can see that when the γ equal to 0.6, the testing performance of the model is the best on multiple datasets. This is because when increasing $\gamma = 1$, the exponential term tends to degenerate into the mean absolute error loss, and when γ approaches 0, the exponential term is prone to degenerate into the cross entropy loss function. And a value approaching the middle ($\gamma = 0.6$) balances the two effects well, benefiting from the average absolute error loss and cross-entropy loss, thereby enhancing robustness against noisy labels.

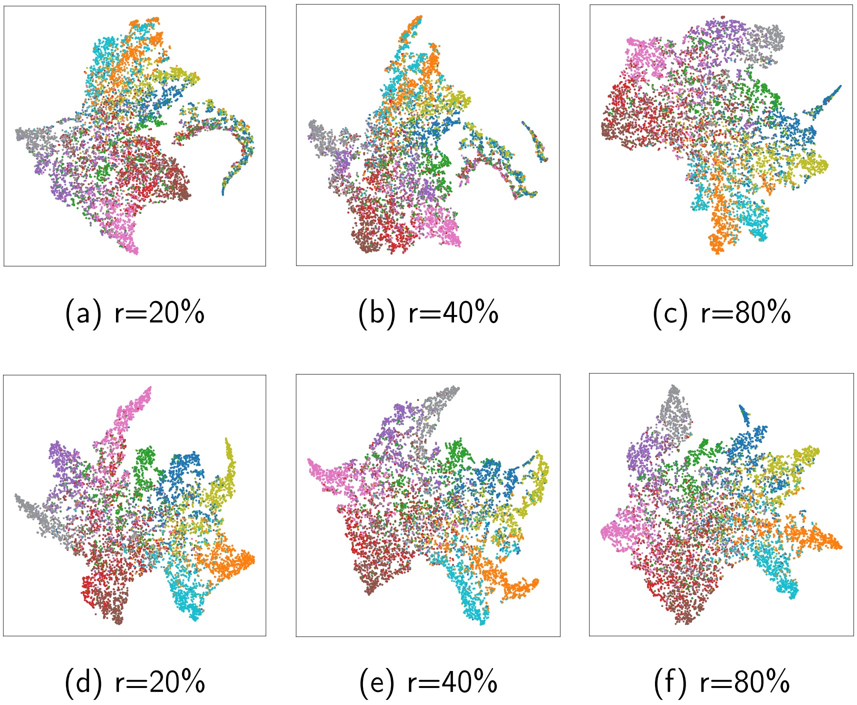
4.6. Evaluation of model representation in latent space

To evaluate the effectiveness of FedCR, we used the dimensionality reduction algorithm t-SNE [89] to visualize the representations in the latent space on the CIFAR-10 test dataset. We present the t-SNE visualizations of the FedCR and FedCon models' feature representations under three noise levels: 20%, 40%, and 80%. The visualizations show the clustering behavior and separability of different classes based on the learned feature embeddings. As shown in Fig. 10, FedCR consistently outperforms FedCon across all noise levels and datasets, which validates that our method can obtain better generalized representations. The key reason for this is FedCR's noisy label filtering mechanism, which refine feature representations by using contrastive learning and generalized cross entropy loss to reduce the impact of noise labels. This helps FedCR maintain clearer decision boundaries between different class representations even as the noise level increases. In contrast, FedCon exhibits relatively poor representation distribution under noisy label conditions, with overlapping features across multiple classes. FedCR enhances the model's generalization ability and improves representation distribution by absorbing local unsupervised knowledge and suppressing noise labeled information. This result meets our expectations. The t-SNE plots reveal that while both FedCR and FedCon can learn reasonable embeddings under low noise conditions (20%), FedCR's clusters are generally tighter and more distinct. As noise increases, FedCR continues to produce well-separated clusters, while FedCon's clusters become increasingly scattered, with more overlaps and less meaningful separability. In scenarios with high noise (80%), FedCR shows remarkable robustness, maintaining noticeable separability between classes, while FedCon's clusters overlap substantially. This shows that FedCR is better suited for handling highly noisy environments. These results provide strong visual evidence supporting the superiority of FedCR in noisy label environments, making it a more effective solution for federated semi-supervised learning tasks with noisy data.

4.7. Discussion and limitations

This paper proposes a simple and effective framework called FedCR, which utilizes the proposed contrastive representation loss and a well-curated unsupervised learning method to address the issues of client label scarcity and noisy labels. Theoretical analysis and empirical experiments have demonstrated significant improvements in federated semi-supervised learning through theoretical analysis and empirical experiments.

However, there are also a few limitations of our approach. First of all, in the experimental setup, we assume that the server side has a small amount of noisy labeled data, and in reality there may be cases where the labels are noisy or the data is completely unlabeled. These conditions are difficult to fulfill and beyond the scope of our study, and are not applicable to our approach. Secondly, we assume


(a) r=20% (b) r=40% (c) r=80%

(d) r=20% (e) r=40% (f) r=80%

Fig. 10. Visualization results of hidden vector features extracted by t-SNE from different models on Cifar-10 test data under three different noise levels. Different colors represent different labels. Each data point represents a data sample in the test set.

that all clients have homogeneous models. However, in the real world, there may be clients with various resources and hardware facilities to participate in federated training, which inevitably results in variations in model capacity or structure. Therefore, our method is not suitable for federated learning with heterogeneous models. Third, our method assumes that the server is completely trusted and the data transmission channel is secure and reliable. Therefore, if the communication channel or server is subjected to illegal attacks and malicious monitoring, it may cause the leakage of client information. Finally, compared with the traditional federated learning algorithm, FedCR slightly introduces the computation and storage costs of the local client due to the incorporation of contrastive learning and the protective model update strategy, which may be difficult to be applied on some devices with low computing power. These limitations provide valuable guidance for future work to improve the generality and practicality of FedCR in various federated learning settings.

5. Conclusion and future work

In this work, we introduce a novel federated semi-supervised learning approach, FedCR, in which the labeled data is solely accessible from the server side while the client-side is provided with entirely unlabeled data. The FedCR employs contrastive representation learning to coordinate supervised learning and unsupervised learning while utilizing the contrastive regularization function and the generalized cross-entropy loss function to train a noise-robust model. In addition, we adopt a dedicated strategy for updating our models to protect data privacy, while also realizing savings in computing and storage expenses. Evaluation of the proposed method on several image datasets under IID and non-IID settings demonstrates the superiority of FedCR over existing federated semi-supervised learning algorithms. Furthermore, the model performs well even when exposed to noise interference at various levels. Our future work aims to extend the FedCR algorithm to specific tasks such as medical image annotation while continuing to refine the model for optimal application in its intended scenarios, taking into account the aforementioned limitations.

CRediT authorship contribution statement

Wenjie Mao: Writing – review & editing, Visualization, Validation, Supervision, Software, Methodology, Formal analysis. **Bin Yu:** Resources, Funding acquisition. **Yihan Lv:** Writing – review & editing, Writing – original draft, Visualization, Methodology, Formal analysis. **Yu Xie:** Writing – review & editing, Visualization, Methodology. **Chen Zhang:** Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors wish to thank the editors and anonymous reviewers for their valuable comments and helpful suggestions which greatly improved the paper's quality. This work was supported by the Key Research and Development Program of Shaanxi, China (Program No. 2022ZDLGY01-13), the National Natural Science Foundation of China (Grant No. 62472270, 62106131, 62101399), the National Key Research and Development Program of China (Grant No. 2020YFC1523304), Research Project Supported by Shanxi Scholarship Council of China (Grant No. 2022-007).

Data availability

Data will be made available on request.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, B.A. y Arcas, Communication-efficient learning of deep networks from decentralized data, in: Artificial Intelligence and Statistics, PMLR, 2017, pp. 1273–1282.

[2] T. Li, A.K. Sahu, A. Talwalkar, V. Smith, Federated learning: Challenges, methods, and future directions, IEEE Signal Process. Mag. 37 (3) (2020) 50–60.

[3] H. Guan, P.-T. Yap, A. Bozoki, M. Liu, Federated learning for medical image analysis: A survey, Pattern Recognit. (2024) 110424.

[4] Y. Liu, A. Huang, Y. Luo, H. Huang, Y. Liu, Y. Chen, L. Feng, T. Chen, H. Yu, Q. Yang, Fedvision: An online visual object detection platform powered by federated learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2020.

[5] T. Kim, E. Lin, J. Lee, C. Lau, V. Mugunthan, Navigating data heterogeneity in federated learning: A semi-supervised approach for object detection, Adv. Neural Inf. Process. Syst. 36 (2024).

[6] Y. Jiang, B. Ma, X. Wang, G. Yu, P. Yu, Z. Wang, W. Ni, R.P. Liu, Blockchained federated learning for internet of things: A comprehensive survey, ACM Comput. Surv. 56 (10) (2024) 1–37.

[7] W. Jeong, J. Yoon, E. Yang, S.J. Hwang, Federated semi-supervised learning with inter-client consistency & disjoint learning, in: 9th International Conference on Learning Representations, ICLR 2021, International Conference on Learning Representations, ICLR, 2021.

[8] Z. Zhang, Y. Yang, Z. Yao, Y. Yan, J.E. Gonzalez, K. Ramchandran, M.W. Mahoney, Improving semi-supervised federated learning by reducing the gradient diversity of models, in: 2021 IEEE International Conference on Big Data (Big Data), IEEE, 2021, pp. 1214–1225.

[9] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, Y. Khazaeni, Federated learning with matched averaging, 2020, arXiv preprint arXiv:2002.06440.

[10] T. Li, M. Sanjabi, A. Beirami, V. Smith, Fair resource allocation in federated learning, 2019, arXiv preprint arXiv:1905.10497.

[11] X.-X. Wei, H. Huang, Balanced federated semisupervised learning with fairness-aware pseudo-labeling, IEEE Trans. Neural Netw. Learn. Syst. (2023).

[12] T. Xiao, T. Xia, Y. Yang, C. Huang, X. Wang, Learning from massive noisy labeled data for image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.

[13] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning (still) requires rethinking generalization, Proc. Commun. ACM (2021).

[14] Y. Liu, Y. Liu, B.X. Yu, S. Zhong, Z. Hu, Noise-robust oversampling for imbalanced data classification, Pattern Recognit. (2023).

[15] Z. Lu, H. Pan, Y. Dai, X. Si, Y. Zhang, Federated learning with non-iid data: A survey, IEEE Internet Things J. (2024).

[16] B. Yu, W. Mao, Y. Lv, C. Zhang, Y. Xie, A survey on federated learning in data mining, Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. 12 (1) (2022) e1443.

[17] Z. Zhao, J. Wang, W. Hong, T.Q. Quek, Z. Ding, M. Peng, Ensemble federated learning with non-IID data in wireless networks, IEEE Trans. Wirel. Commun. (2023).

[18] X. Li, H. Zhao, W. Deng, IOFL: Intelligent optimization-based federated learning for non-IID data, IEEE Internet Things J. (2024).

[19] W. Mao, B. Yu, C. Zhang, A. Qin, Y. Xie, FedKT: Federated learning with knowledge transfer for non-IID data, Pattern Recognit. 159 (2025) 111143.

[20] T. Tuor, S. Wang, B.J. Ko, C. Liu, K.K. Leung, Overcoming noisy and irrelevant data in federated learning, in: 2020 25th International Conference on Pattern Recognition, ICPR, IEEE, 2021, pp. 5020–5027.

[21] S. Itahara, T. Nishio, Y. Koda, M. Morikura, K. Yamamoto, Distillation-based semi-supervised federated learning for communication-efficient collaborative training with non-iid private data, IEEE Trans. Mob. Comput. 22 (1) (2021) 191–205.

[22] C. Zhang, Y. Xie, T. Chen, W. Mao, B. Yu, Prototype similarity distillation for communication-efficient federated unsupervised representation learning, IEEE Trans. Knowl. Data Eng. (2024).

[23] M. Duan, D. Liu, X. Chen, Y. Tan, J. Ren, L. Qiao, L. Liang, Astraea: Self-balancing federated learning for improving classification accuracy of mobile deep learning applications, in: Proceedings of the 2019 IEEE 37th International Conference on Computer Design, ICCD, 2019.

[24] T. Yoon, S. Shin, S.J. Hwang, E. Yang, Fedmix: Approximation of mixup under mean augmented federated learning, 2021, arXiv preprint arXiv:2107.00233.

[25] C. T. Dinh, N. Tran, J. Nguyen, Personalized federated learning with moreau envelopes, Adv. Neural Inf. Process. Syst. 33 (2020) 21394–21405.

[26] L. Collins, H. Hassani, A. Mokhtari, S. Shakkottai, Exploiting shared representations for personalized federated learning, in: International Conference on Machine Learning, PMLR, 2021, pp. 2089–2099.

[27] F. Sabah, Y. Chen, Z. Yang, M. Azam, N. Ahmad, R. Sarwar, Model optimization techniques in personalized federated learning: A survey, Expert Syst. Appl. 243 (2024) 122874.

[28] A. Ahmad, W. Luo, A. Robles-Kelly, Robust federated learning under statistical heterogeneity via hessian spectral decomposition, Pattern Recognit. 141 (2023) 109635.

[29] E. Yu, Z. Ye, Z. Zhang, L. Qian, M. Xie, A federated recommendation algorithm based on user clustering and meta-learning, Appl. Soft Comput. 158 (2024) 111483.

[30] H. Lin, J. Lou, L. Xiong, C. Shahabi, Semifed: Semi-supervised federated learning with consistency and pseudo-labeling, 2021, arXiv preprint arXiv:2108.09412.

[31] W. Kim, K. Park, K. Sohn, R. Shu, H.-S. Kim, Federated semi-supervised learning with prototypical networks, 2022, arXiv preprint arXiv:2205.13921.

[32] D. Yang, Z. Xu, W. Li, A. Myronenko, H.R. Roth, S. Harmon, S. Xu, B. Turkbey, E. Turkbey, X. Wang, et al., Federated semi-supervised learning for COVID region segmentation in chest CT using multi-national data from China, Italy, Japan, Med. Image Anal. 70 (2021) 101992.

[33] Q. Liu, H. Yang, Q. Dou, P.-A. Heng, Federated semi-supervised medical image classification via inter-client relation matching, in: Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part III 24, Springer, 2021, pp. 325–335.

[34] X. Liang, Y. Lin, H. Fu, L. Zhu, X. Li, RSCFed: random sampling consensus federated semi-supervised learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.

[35] M. Li, Q. Li, Y. Wang, Class balanced adaptive pseudo labeling for federated semi-supervised learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 16292–16301.

[36] C. Fan, J. Hu, J. Huang, Private semi-supervised federated learning, in: IJCAI, 2022, pp. 2009–2015.

[37] L. Gao, H. Fu, L. Li, Y. Chen, M. Xu, C.-Z. Xu, Feddc: Federated learning with non-iid data via local drift decoupling and correction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10112–10121.

[38] Q. Guo, D. Wu, Y. Qi, S. Qi, Dual class-aware contrastive federated semi-supervised learning, IEEE Trans. Mob. Comput. (2024).

[39] L. Qiu, J. Cheng, H. Gao, W. Xiong, H. Ren, Federated semi-supervised learning for medical image segmentation via pseudo-label denoising, IEEE J. Biomed. Heal. Inform. 27 (10) (2023) 4672–4683.

[40] Y. Liu, X. Yuan, R. Zhao, Y. Zheng, Y. Zheng, RC-SSFL: Towards robust and communication-efficient semi-supervised federated learning system, 2020, arXiv preprint arXiv:2012.04432.

[41] Z. Long, J. Wang, Y. Wang, H. Xiao, F. Ma, Fedcon: A contrastive framework for federated semi-supervised learning, 2021, arXiv preprint arXiv:2109.04533.

[42] E. Diao, J. Ding, V. Tarokh, Semifl: Semi-supervised federated learning for unlabeled clients with alternate training, Adv. Neural Inf. Process. Syst. 35 (2022) 17871–17884.

[43] Z. Zhong, J. Wang, W. Bao, J. Zhou, X. Zhu, X. Zhang, Semi-HFL: semi-supervised federated learning for heterogeneous devices, Complex Intell. Syst. 9 (2) (2023) 1995–2017.

[44] X. Qiu, Y. Gao, L. Sani, H. Pan, W. Zhao, P.P. Gusmao, M. Alibeigi, A. Iacob, N.D. Lane, FedAnchor: Enhancing federated semi-supervised learning with label contrastive loss for unlabeled clients, 2024, arXiv preprint arXiv:2402.10191.

[45] S. Lee, T.-L.V. Le, J. Shin, S.-J. Lee, FL2: Overcoming few labels in federated semi-supervised learning, 2024, arXiv preprint arXiv:2410.23227.

[46] L. Che, Z. Long, J. Wang, Y. Wang, H. Xiao, F. Ma, FedTriNet: A pseudo labeling method with three players for federated semi-supervised learning, in: 2021 IEEE International Conference on Big Data (Big Data), IEEE, 2021, pp. 715–724.

[47] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick, Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020.

[48] G. Chae, J. Lee, S.B. Kim, Contrastive learning with hard negative samples for chest X-ray multi-label classification, Appl. Soft Comput. 165 (2024) 112101.

[49] W. Zhang, Y. Lin, Y. Liu, H. You, P. Wu, F. Lin, X. Zhou, Self-supervised reinforcement learning with dual-reward for knowledge-aware recommendation, Appl. Soft Comput. 131 (2022) 109745.

[50] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: Proceedings of the PMLR International Conference on Machine Learning, 2020.

[51] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, et al., Bootstrap your own latent-a new approach to self-supervised learning, Proc. Adv. Neural Inf. Process. Syst. (2020).

[52] X. Chen, K. He, Exploring simple siamese representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.

[53] Y. Jin, Y. Liu, K. Chen, Q. Yang, Federated learning without full labels: A survey, 2023, arXiv preprint arXiv:2303.14453.

[54] W. Zhuang, X. Gan, Y. Wen, S. Zhang, S. Yi, Collaborative unsupervised visual representation learning from decentralized data, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 4912–4921.

[55] C. He, Z. Yang, E. Mushtaq, S. Lee, M. Soltanolkotabi, S. Avestimehr, Ssfl: Tackling label deficiency in federated learning via personalized self-supervision, 2021, arXiv preprint arXiv:2110.02470.

[56] Y.A.U. Rehman, Y. Gao, P.P.B. De Gusmão, M. Alibeigi, J. Shen, N.D. Lane, L-dawa: Layer-wise divergence aware weight aggregation in federated self-supervised visual representation learning, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 16464–16473.

[57] S. Li, Y. Mao, J. Li, Y. Xu, J. Li, X. Chen, S. Liu, X. Zhao, FedUTN: federated self-supervised learning with updating target network, Appl. Intell. 53 (9) (2023) 10879–10892.

[58] X. Liao, W. Liu, C. Chen, P. Zhou, F. Yu, H. Zhu, B. Yao, T. Wang, X. Zheng, Y. Tan, Rethinking the representation in federated unsupervised learning with non-IID data, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 22841–22850.

[59] M. Li, X. Zhang, Q. Wang, T. LIU, R. Wu, W. Wang, F. Zhuang, H. Xiong, D. Yu, Resource-aware federated self-supervised learning with global class representations, in: The Thirty-Eighth Annual Conference on Neural Information Processing Systems, 2024, URL https://openreview.net/forum?id=Of4iNAIUSe.

[60] H. Song, M. Kim, D. Park, Y. Shin, J.-G. Lee, Learning from noisy labels with deep neural networks: A survey, Proc. IEEE Trans. Neural Netw. Learn. Syst. (2022).

[61] X. Jiang, S. Sun, Y. Wang, M. Liu, Towards federated learning against noisy labels via local self-regularization, in: Proceedings of the 31st ACM International Conference on Information & Knowledge Management, 2022, pp. 862–873.

[62] Z. Zhang, M. Sabuncu, Generalized cross entropy loss for training deep neural networks with noisy labels, Proc. Adv. Neural Inf. Process. Syst. (2018).

[63] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, J. Bailey, Symmetric cross entropy for robust learning with noisy labels, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019.

[64] Y. Liu, H. Guo, Peer loss functions: Learning from noisy labels without knowing noise rates, in: Proceedings of the PMLR International Conference on Machine Learning, 2020.

[65] Y. Kim, J. Yim, J. Yun, J. Kim, Nlnl: Negative learning for noisy labels, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019.

[66] L. Yi, S. Liu, Q. She, A.I. McLeod, B. Wang, On learning contrastive representations for learning with noisy labels, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022.

[67] A. Ghosh, A. Lan, Contrastive learning improves model robustness under label noise, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021.

[68] N. Karim, U. Khalid, A. Esmaeili, N. Rahnavard, Cnll: A semi-supervised approach for continual noisy label learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 3878–3888.

[69] Y. Zhang, X. Zhang, J. Li, R.C. Qiu, H. Xu, Q. Tian, Semi-supervised contrastive learning with similarity co-calibration, IEEE Trans. Multimed. 25 (2022) 1749–1759.

[70] B. Huang, Y. Lin, C. Xu, Contrastive label correction for noisy label learning, Inform. Sci. 611 (2022) 173–184.

[71] J. Xu, Z. Chen, T.Q. Quek, K.F.E. Chong, Fedcorr: Multi-stage federated learning for label noise correction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10184–10193.

[72] S. Yang, H. Park, J. Byun, C. Kim, Robust federated learning with noisy labels, IEEE Intell. Syst. 37 (2) (2022) 35–43.

[73] V. Tsouvalas, A. Saeed, T. Özçelebi, N. Meratnia, Federated learning with noisy labels: Achieving generalization in the face of label noise, in: First Workshop on Interpolation Regularizers and beyond At NeurIPS 2022, 2022.

[74] X. Jiang, S. Sun, J. Li, J. Xue, R. Li, Z. Wu, G. Xu, Y. Wang, M. Liu, Tackling noisy clients in federated learning with end-to-end label correction, in: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, 2024, pp. 1015–1026.

[75] L. Wang, J. Bian, J. Xu, Federated learning with instance-dependent noisy label, in: ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, IEEE, 2024, pp. 8916–8920.

[76] M. Laskin, A. Srinivas, P. Abbeel, Curl: Contrastive unsupervised representations for reinforcement learning, in: Proceedings of the PMLR International Conference on Machine Learning, 2020.

[77] E. Xie, J. Ding, W. Wang, X. Zhan, H. Xu, P. Sun, Z. Li, P. Luo, Detco: Unsupervised contrastive learning for object detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021.

[78] Y. Xue, K. Whitecross, B. Mirzasoleiman, Investigating why contrastive learning benefits robustness against label noise, in: International Conference on Machine Learning, PMLR, 2022, pp. 24851–24871.

[79] M. Li, M. Soltanolkotabi, S. Oymak, Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks, in: International Conference on Artificial Intelligence and Statistics, PMLR, 2020, pp. 4313–4324.

[80] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, A.Y. Ng, et al., Reading digits in natural images with unsupervised feature learning, in: NIPS Workshop on Deep Learning and Unsupervised Feature Learning, Vol. 2011, Granada, 2011, p. 4.

[81] A. Krizhevsky, G. Hinton, et al., Learning multiple layers of features from tiny images, 2009.

[82] P. Chen, G. Chen, J. Ye, P.-A. Heng, et al., Noise against noise: stochastic label noise helps combat inherent label noise, in: Proceedings of the International Conference on Learning Representations, 2021.

[83] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C.A. Raffel, E.D. Cubuk, A. Kurakin, C.-L. Li, Fixmatch: Simplifying semi-supervised learning with consistency and confidence, Proc. Adv. Neural Inf. Process. Syst. (2020).

[84] T. Li, A.K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, V. Smith, Federated optimization in heterogeneous networks, Proc. Mach. Learn. Syst. 2 (2020) 429–450.

[85] Q. Xie, Z. Dai, E. Hovy, T. Luong, Q. Le, Unsupervised data augmentation for consistency training, Proc. Adv. Neural Inf. Process. Syst. (2020).

[86] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: Proceedings of the ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14, 2016.

[87] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.

[88] W. Zhuang, Y. Wen, S. Zhang, Divergence-aware federated self-supervised learning, 2022, arXiv preprint arXiv:2204.04385.

[89] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008).